THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOSS PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

© 2015–2019 Cisco Systems, Inc. All rights reserved.
# CONTENTS

## PREFACE

- Preface xvii
  - Changes to This Document xvii
  - Communications, Services, and Additional Information xvii

## CHAPTER 1

- IS-IS Commands 1
  - address-family (IS-IS) 4
  - adjacency stagger 5
  - adjacency-check disable 7
  - circuit-type 8
  - clear isis process 10
  - clear isis route 11
  - clear isis statistics 12
  - csnp-interval 13
  - default-information originate (IS-IS) 14
  - disable (IS-IS) 16
  - distance (IS-IS) 17
  - hello-interval (IS-IS) 19
  - hello-multiplier 21
  - hello-padding 23
  - hello-password 24
  - hello-password keychain 26
  - hello-password accept 27
  - hostname dynamic disable 28
  - ignore-lsp-errors 29
  - instance-id 30
  - interface (IS-IS) 31
Routing Command Reference for Cisco NCS 5000 Series Routers

Contents

ispf 32
is-type 33
log pdu drops 35
lsp fast-flood threshold 36
lsp-gen-interval 37
lsp-interval 39
lsp-mtu 40
lsp-password 41
lsp-password accept 43
lsp-refresh-interval 44
maximum-paths (IS-IS) 45
maximum-redistributed-prefixes (IS-IS) 46
max-lsp-lifetime 47
mesh-group (IS-IS) 48
metric (IS-IS) 50
metric-style narrow 52
metric-style transition 53
metric-style wide 54
min-lsp-arrivaltime 56
mpls ldp auto-config 58
mpls ldp synce (IS-IS) 59
nsf (IS-IS) 60
nsf interface-expires 61
nsf interface-timer 62
nsf lifetime (IS-IS) 63
passive (IS-IS) 64
point-to-point 65
prefix-sid index 66
priority (IS-IS) 67
propagate level 68
redistribute (IS-IS) 69
retransmit-interval (IS-IS) 73
retransmit-throttle-interval 74
route source first-hop 75
segment-routing 76
set-attached-bit 77
set-overload-bit 79
show isis 81
show isis adjacency 83
show isis adjacency-log 85
show isis checkpoint adjacency 87
show isis checkpoint interface 89
show isis checkpoint lsp 90
show isis database 92
show isis database-log 101
show isis hostname 103
show isis interface 105
show isis lsp-log 109
show isis mesh-group 111
show isis neighbors 113
show isis protocol 116
show isis route 118
show isis spf-log 121
show isis statistics 128
show isis topology 132
show protocols (IS-IS) 135
shutdown (IS-IS) 138
single-topology 139
snmp-server traps isis 140
spf-interval 142
spf prefix-priority (IS-IS) 144
summary-prefix (IS-IS) 146
suppressed 148
tag (IS-IS) 149
topology-id 150
trace (IS-IS) 151

CHAPTER 2  OSPF Commands 153
address-family (OSPF) 156
adjacency stagger 157
area (OSPF) 159
authentication (OSPF) 160
authentication-key (OSPF) 162
auto-cost (OSPF) 164
capability opaque disable 165
clear ospf process 166
clear ospf redistribution 167
clear ospf routes 168
clear ospf statistics 169
cost (OSPF) 170
cost-fallback (OSPF) 172
database-filter all out (OSPF) 174
dead-interval (OSPF) 175
default-cost (OSPF) 177
default-information originate (OSPF) 178
default-metric (OSPF) 180
demand-circuit (OSPF) 181
disable-dn-bit-check 182
distance (OSPF) 183
distance ospf 185
distribute-list 187
domain-id (OSPF) 189
domain-tag 191
flood-reduction (OSPF) 192
hello-interval (OSPF) 193
ignore lsa mospf 195
interface (OSPF) 196
log adjacency changes (OSPF) 198
loopback stub-network 199
max-lsa 200
max-metric 202
maximum interfaces (OSPF) 205
maximum paths (OSPF) 206
maximum redistributed-prefixes (OSPF) 207
message-digest-key 208
mpls ldp auto-config (OSPF) 211
mpls ldp sync (OSPF) 212
mtu-ignore (OSPF) 213
multi-area-interface 215
neighbor (OSPF) 217
neighbor database-filter all out 219
network (OSPF) 220
nsf (OSPF) 222
nsf flush-delay-time (OSPF) 224
nsf interval (OSPF) 225
nsf lifetime (OSPF) 226
nssa (OSPF) 227
ospf name-lookup 229
packet-size (OSPF) 230
passive (OSPF) 232
priority (OSPF) 234
protocol shutdown 236
queue dispatch flush-lsa 237
queue dispatch incoming 238
queue dispatch rate-limited-lsa 239
queue dispatch spf-lsa-limit 240
queue limit 241
range (OSPF) 242
redistribute (OSPF) 244
retransmit-interval (OSPF) 248
route-policy (OSPF) 250
router-id (OSPF) 251
router ospf 253
sham-link 254
show ospf 255
show ospf border-routers 257
show ospf database 259
show ospf flood-list 272
show ospf interface 274
show ospf message-queue 276
show ospf neighbor 278
show ospf request-list 285
show ospf retransmission-list 287
show ospf routes 289
show ospf sham-links 294
show ospf summary-prefix 296
show ospf virtual-links 298
show protocols (OSPF) 300
snmp context (OSPF) 302
snmp trap (OSPF) 304
snmp trap rate-limit (OSPF) 305
spf prefix-priority (OSPFv2) 306
stub (OSPF) 308
summary-prefix (OSPF) 309
timers lsa group-pacing 311
timers lsa min-arrival 312
timers throttle lsa all (OSPF) 313
timers throttle spf (OSPF) 315
transmit-delay (OSPF) 317
virtual-link (OSPF) 319
vrf (OSPF) 321

CHAPTER 3

OSPFv3 Commands 323
address-family (OSPFv3) 326
area (OSPFv3) 327
authentication (OSPFv3) 328
auto-cost (OSPFv3) 330
capability vrf-lite (OSPFv3) 331
clear ospfv3 process 332
clear ospfv3 redistribution 333
clear ospfv3 routes 334
clear ospfv3 statistics 335
cost (OSPFv3) 336
database-filter all out (OSPFv3) 337
dead-interval (OSPFv3) 338
default-cost (OSPFv3) 340
default-information originate (OSPFv3) 341
default-metric (OSPFv3) 343
demand-circuit (OSPFv3) 344
distance ospfv3 346
distribute-list prefix-list in 348
distribute-list prefix-list out 350
domain-id (OSPFv3) 352
encryption 353
flood-reduction (OSPFv3) 355
graceful-restart (OSPFv3) 356
hello-interval (OSPFv3) 357
instance (OSPFv3) 359
interface (OSPFv3) 361
log adjacency changes (OSPFv3) 363
maximum interfaces (OSPFv3) 364
maximum paths (OSPFv3) 365
maximum redistributed-prefixes (OSPFv3) 366
mtu-ignore (OSPFv3) 368
neighbor (OSPFv3) 370
network (OSPFv3) 372
nssa (OSPFv3) 374
ospfv3 name-lookup 376
packet-size (OSPFv3) 377
passive (OSPFv3) 378
priority (OSPFv3) 380
protocol shutdown (OSPFv3) 381
range (OSPFv3) 382
redistribute (OSPFv3) 384
CHAPTER 4  RIB Commands  465

- address-family next-hop dampening disable  467
- clear route  468
- maximum prefix (RIB)  470
- lcc  471
- rcc  472
- recursion-depth-max  473
- router rib  474
- rump always-replicate  475
- show lcc statistics  476
- show rcc  478
- show rcc statistics  480
- show rib  482
- show rib afi-all  484
- show rib attributes  486
- show rib client-id  487
- show rib clients  489
- show rib extcomms  491
- show rib firsthop  493
- show rib history  495
- show rib next-hop  497
- show rib opaques  499
- show rib protocols  501
- show rib recursion-depth-max  503
- show rib statistics  505
- show rib tables  507
- show rib trace  509
- show rib vpn-attributes  511
- show route  513
- show route backup  520
- show route best-local  523
- show route connected  525
- show route local  527
<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>show route longer-prefixes</td>
<td>529</td>
</tr>
<tr>
<td>show route next-hop</td>
<td>531</td>
</tr>
<tr>
<td>show route quarantined</td>
<td>533</td>
</tr>
<tr>
<td>show route resolving-next-hop</td>
<td>535</td>
</tr>
<tr>
<td>show route static</td>
<td>537</td>
</tr>
<tr>
<td>show route summary</td>
<td>539</td>
</tr>
</tbody>
</table>

**CHAPTER 5**

**Routing Policy Language Commands** 543

- abort (RPL) 547
- add 549
- apply 550
- as-path in 552
- as-path is-local 554
- as-path length 555
- as-path neighbor-is 556
- as-path originates-from 558
- as-path passes-through 560
- as-path-set 562
- as-path unique-length 564
- community is-empty 566
- community matches-any 567
- community matches-every 569
- community-set 571
- delete community 574
- delete extcommunity rt 576
- destination in 578
- done 580
- drop 582
- edit 583
- end-global 586
- end-policy 587
- end-set 588
- extcommunity rt is-empty 590
- extcommunity rt matches-any 591
<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>rpl maximum</td>
<td>649</td>
</tr>
<tr>
<td>rpl set-exit-as-abort</td>
<td>650</td>
</tr>
<tr>
<td>set administrative-distance</td>
<td>651</td>
</tr>
<tr>
<td>set aigp-metric</td>
<td>652</td>
</tr>
<tr>
<td>set community</td>
<td>653</td>
</tr>
<tr>
<td>set core-tree</td>
<td>655</td>
</tr>
<tr>
<td>set dampening</td>
<td>656</td>
</tr>
<tr>
<td>set extcommunity cost</td>
<td>658</td>
</tr>
<tr>
<td>set extcommunity rt</td>
<td>660</td>
</tr>
<tr>
<td>set ip-precedence</td>
<td>662</td>
</tr>
<tr>
<td>set isis-metric</td>
<td>663</td>
</tr>
<tr>
<td>set label</td>
<td>664</td>
</tr>
<tr>
<td>set label-mode</td>
<td>665</td>
</tr>
<tr>
<td>set level</td>
<td>667</td>
</tr>
<tr>
<td>set local-preference</td>
<td>668</td>
</tr>
<tr>
<td>set med</td>
<td>669</td>
</tr>
<tr>
<td>set metric-type (IS-IS)</td>
<td>671</td>
</tr>
<tr>
<td>set metric-type (OSPF)</td>
<td>673</td>
</tr>
<tr>
<td>set next-hop</td>
<td>674</td>
</tr>
<tr>
<td>set origin</td>
<td>676</td>
</tr>
<tr>
<td>set ospf-metric</td>
<td>677</td>
</tr>
<tr>
<td>set path-selection</td>
<td>678</td>
</tr>
<tr>
<td>set qos-group (RPL)</td>
<td>679</td>
</tr>
<tr>
<td>set rib-metric</td>
<td>680</td>
</tr>
<tr>
<td>set rip-metric</td>
<td>681</td>
</tr>
<tr>
<td>set rip-tag</td>
<td>682</td>
</tr>
<tr>
<td>set rpf-topology</td>
<td>683</td>
</tr>
<tr>
<td>set spf-priority</td>
<td>685</td>
</tr>
<tr>
<td>set tag</td>
<td>686</td>
</tr>
<tr>
<td>set traffic-index</td>
<td>687</td>
</tr>
<tr>
<td>set vpn-distinguisher</td>
<td>689</td>
</tr>
<tr>
<td>set weight</td>
<td>690</td>
</tr>
<tr>
<td>show rpl</td>
<td>691</td>
</tr>
<tr>
<td>show rpl active as-path-set</td>
<td>693</td>
</tr>
</tbody>
</table>
show rpl active community-set 695
show rpl active extcommunity-set 697
show rpl active prefix-set 700
show rpl active rd-set 702
show rpl active route-policy 704
show rpl as-path-set 706
show rpl as-path-set attachpoints 707
show rpl as-path-set references 709
show rpl community-set 711
show rpl community-set attachpoints 713
show rpl community-set references 715
show rpl extcommunity-set 717
show rpl inactive as-path-set 720
show rpl inactive community-set 722
show rpl inactive extcommunity-set 724
show rpl inactive prefix-set 726
show rpl inactive rd-set 728
show rpl inactive route-policy 730
show rpl maximum 732
show rpl policy-global references 734
show rpl prefix-set 736
show rpl prefix-set attachpoints 737
show rpl prefix-set references 739
show rpl rd-set 741
show rpl rd-set attachpoints 742
show rpl rd-set references 744
show rpl route-policy 746
show rpl route-policy attachpoints 749
show rpl route-policy inline 751
show rpl route-policy references 753
show rpl route-policy uses 756
show rpl unused as-path-set 759
show rpl unused community-set 762
show rpl unused extcommunity-set 765
show rpl unused prefix-set 766
show rpl unused rd-set 769
show rpl unused route-policy 770
source in 773
suppress-route 775
tag 776
tag in 777
tag-set 779
unsuppress-route 780
vpn-distinguisher is 781

CHAPTER 6

Static Routing Commands 783
address-family (static) 784
maximum path (static) 785
route (static) 786
router static 788
Preface

The *Routing Command Reference for Cisco NCS 5000 Series Routers* preface contains these sections:

- **Changes to This Document, on page xvii**
- **Communications, Services, and Additional Information, on page xvii**

### Changes to This Document

This table lists the technical changes made to this document since it was first printed.

<table>
<thead>
<tr>
<th>Date</th>
<th>Change Summary</th>
</tr>
</thead>
<tbody>
<tr>
<td>December 2015</td>
<td>Initial release of this document.</td>
</tr>
<tr>
<td>November 2016</td>
<td>Republished with documentation updates for 6.1.1 features.</td>
</tr>
<tr>
<td>July 2017</td>
<td>Republished with documentation updates for Release 6.2.2 features.</td>
</tr>
<tr>
<td>September 2017</td>
<td>Republished with documentation updates for Release 6.3.1 features.</td>
</tr>
<tr>
<td>July 2018</td>
<td>Republished with documentation updates for Release 6.5.1 features.</td>
</tr>
</tbody>
</table>

### Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at [Cisco Profile Manager](#).
- To get the business impact you’re looking for with the technologies that matter, visit [Cisco Services](#).
- To submit a service request, visit [Cisco Support](#).
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit [Cisco Marketplace](#).
- To obtain general networking, training, and certification titles, visit [Cisco Press](#).
To find warranty information for a specific product or product family, access Cisco Warranty Finder.

**Cisco Bug Search Tool**

Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.
IS-IS Commands

This module describes the commands used to configure and monitor the Intermediate System-to-Intermediate System (IS-IS) protocol on Cisco NCS 5000 Series Routers.

For detailed information about IS-IS concepts, configuration tasks, and examples, see the Implementing IS-IS on Routing Command Reference for Cisco NCS 5000 Series Routers module in the Routing Configuration Guide for Cisco NCS 5000 Series Routers.

Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

- address-family (IS-IS), on page 4
- adjacency stagger, on page 5
- adjacency-check disable, on page 7
- circuit-type, on page 8
- clear isis process, on page 10
- clear isis route, on page 11
- clear isis statistics, on page 12
- csnp-interval, on page 13
- default-information originate (IS-IS), on page 14
- disable (IS-IS), on page 16
- distance (IS-IS), on page 17
- hello-interval (IS-IS), on page 19
- hello-multiplier, on page 21
- hello-padding, on page 23
- hello-password, on page 24
- hello-password keychain, on page 26
- hello-password accept, on page 27
- hostname dynamic disable, on page 28
- ignore-lsp-errors, on page 29
- instance-id, on page 30
- interface (IS-IS), on page 31
- ispf, on page 32
- is-type, on page 33
- log pdu drops, on page 35
- lsp fast-flood threshold, on page 36
- lsp-gen-interval, on page 37
- lsp-interval, on page 39
- lsp-mtu, on page 40
- lsp-password, on page 41
- lsp-password accept, on page 43
- lsp-refresh-interval, on page 44
- maximum-paths (IS-IS), on page 45
- maximum-redistributed-prefixes (IS-IS), on page 46
- max-lsp-lifetime, on page 47
- mesh-group (IS-IS), on page 48
- metric (IS-IS), on page 50
- metric-style narrow, on page 52
- metric-style transition, on page 53
- metric-style wide, on page 54
- min-lsp-arrivaltime, on page 56
- mpls ldp auto-config, on page 58
- mpls ldp sync (IS-IS), on page 59
- nsf (IS-IS), on page 60
- nsf interface-expires, on page 61
- nsf interface-timer, on page 62
- nsf lifetime (IS-IS), on page 63
- passive (IS-IS), on page 64
- point-to-point, on page 65
- prefix-sid index, on page 66
- priority (IS-IS), on page 67
- propagate level, on page 68
- redistribute (IS-IS), on page 69
- retransmit-interval (IS-IS), on page 73
- retransmit-throttle-interval, on page 74
- route source first-hop, on page 75
- segment-routing, on page 76
- set-attached-bit, on page 77
- set-overload-bit, on page 79
- show isis, on page 81
- show isis adjacency, on page 83
- show isis adjacency-log, on page 85
- show isis checkpoint adjacency, on page 87
- show isis checkpoint interface, on page 89
- show isis checkpoint lsp, on page 90
- show isis database, on page 92
- show isis database-log, on page 101
- show isis hostname, on page 103
- show isis interface, on page 105
- show isis lsp-log, on page 109
• show isis mesh-group, on page 111
• show isis neighbors, on page 113
• show isis protocol, on page 116
• show isis route, on page 118
• show isis spf-log, on page 121
• show isis statistics, on page 128
• show isis topology, on page 132
• show protocols (IS-IS), on page 135
• shutdown (IS-IS), on page 138
• single-topology, on page 139
• snmp-server traps isis, on page 140
• spf-interval, on page 142
• spf prefix-priority (IS-IS), on page 144
• summary-prefix (IS-IS), on page 146
• suppressed, on page 148
• tag (IS-IS), on page 149
• topology-id, on page 150
• trace (IS-IS), on page 151
address-family (IS-IS)

To enter address family configuration mode for configuring Intermediate System-to-Intermediate System (IS-IS) routing that use standard IP Version 4 (IPv4) and IP Version 6 (IPv6) address prefixes, use the **address-family** command in XR Config mode or interface configuration mode. To disable support for an address family, use the **no** form of this command.

```
address-family {ipv4|ipv6} {unicast}
no address-family {ipv4|ipv6} {unicast}
```

**Syntax Description**

- **ipv4**  
  Specifies IPv4 address prefixes.
- **ipv6**  
  Specifies IPv6 address prefixes.
- **unicast**  
  Specifies unicast address prefixes.

**Command Default**

An address family is not specified. The default subaddress family (SAFI) is unicast.

**Command Modes**

- XR Config mode
- Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **address family** command to place the router or interface in address family configuration mode. In router address family configuration mode, you can configure routing that uses standard IPv4 or IPv6 address prefixes. An address family must be specified in interface configuration mode. In interface address family configuration mode, you can alter interface parameters for IPv4 or IPv6.

You must specify an address family in order to configure parameters that pertain to a single address family.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure the IS-IS router process with IPv4 unicast address prefixes:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface HundredGigE 0/1/0/0
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)#
```
adjacency stagger

To configure staggering of IS-IS adjacency during reload, process restart, and process clear, use the `adjacency stagger` command in router configuration mode. To turn off adjacency staggering, either use the `disable` keyword or use the `no` form of this command.

```
adjacency stagger {disable|initial-num-nbr max-num-nbr}
no adjacency stagger
```

<table>
<thead>
<tr>
<th>disable</th>
<th>Disables adjacency staggering.</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>initial-num-nbr</code></td>
<td>The initial number of simultaneous neighbors allowed to form adjacency to FULL in any area to bring up to FULL after a router reload, IS-IS process restart, or IS-IS process clear. Range is 1-65535. Default is 2.</td>
</tr>
<tr>
<td><code>max-num-nbr</code></td>
<td>The subsequent number of simultaneous neighbors allowed to form adjacency, per IS-IS instance, after the initial set of IS-IS neighbors have become FULL. Range is 1-65535. Default is 64.</td>
</tr>
</tbody>
</table>

**Command Default**

IS-IS adjacency staggering is enabled.

**Command Modes**

Router configuration

**Table 1: Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.3.1</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

To use this command, you must be in a user group associated with a task group that includes appropriate task IDs. If the user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

Staggering of the IS-IS adjacency during reload, process restart (without NSR or graceful-restart), and process clear reduces the overall adjacency convergence time.

Initially, allow 2 (configurable) neighbors to form adjacency to FULL per area. After the first adjacency reaches FULL, up to 64 (configurable) neighbors can form adjacency simultaneously for the IS-IS instance (all areas). However, areas without any FULL adjacency is restricted by the initial area limit.

**Note**

Adjacency stagger and IS-IS nonstop forwarding (NSF) are mutually exclusive. Adjacency stagger is not activated if nonstop forwarding (NSF) is configured in the router along with IS-IS configuration.

**Table 2: Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>IS-IS</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to configure adjacency stagger for a 2 neighbors initially and for a maximum of 32 neighbors:

Router# configure
Router(config)# router isis 1
Router(config-isis)# adjacency stagger 2 32
adjacency-check disable

To suppress Intermediate System-to-Intermediate System (IS-IS) IP Version 4 (IPv4) or IP Version 6 (IPv6) protocol-support consistency checks that are performed prior to forming adjacencies on hello packets, use the adjacency-check disable command in address family configuration mode. To remove this function, use the no form of this command.

adjacency-check disable
no adjacency-check disable

Command Default
Adjacency check is enabled

Command Modes
Address family configuration

Command History
Release Modification
Release 6.0 This command was introduced.

Usage Guidelines
IS-IS performs consistency checks on hello packets and forms an adjacency only with a neighboring router that supports the same set of protocols. A router running IS-IS for both IPv4 and IPv6 does not form an adjacency with a router running IS-IS for IPv4 only.

Use the adjacency-check disable command to suppress the consistency checks for IPv6 IS-IS and allow an IPv4 IS-IS router to form an adjacency with a router running IPv4 IS-IS and IPv6. IS-IS never forms an adjacency between a router running IPv4 IS-IS only and a router running IPv6 only.

In addition, the adjacency-check disable command suppresses the IPv4 or IPv6 subnet consistency check and allows IS-IS to form an adjacency with other routers regardless of whether they have an IPv4 or IPv6 subnet in common.

Task ID
Task ID Operations
isis read, write

Examples
The command in the following example disables the adjacency checks:

The following example shows how the network administrator introduces IPv6 into an existing IPv4 IS-IS network and ensures that the checking of hello packet checks from adjacent neighbors is disabled until all neighbor routers are configured to use IPv6:

RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv6 |ipv4
RP/0/RP0/CPU0:router(config-isis-af)# adjacency-check disable
To configure the type of adjacency used for the Intermediate System-to-Intermediate System (IS-IS) protocol, use the `circuit-type` command in interface configuration mode. To reset the circuit type to Level 1 and Level 2, use the `no` form of this command.

```
circuit-type  {level-1|level-1-2|level-2-only}
no  circuit-type
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>level-1</td>
<td>Establishes only Level 1 adjacencies over an interface.</td>
</tr>
<tr>
<td>level-1-2</td>
<td>Establishes both Level 1 and Level 2 adjacencies, if possible.</td>
</tr>
<tr>
<td>level-2-only</td>
<td>Establishes only Level 2 adjacencies over an interface.</td>
</tr>
</tbody>
</table>

**Command Default**

Default adjacency types are Level 1 and Level 2 adjacencies.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Adjacencies may not be established even if allowed by the `circuit-type` command. The proper way to establish adjacencies is to configure a router as a Level 1, Level 1 and Level 2, or Level 2-only system using the `is-type`, on page 33 command. Only on networking devices that are between areas (Level 1 and Level 2 networking devices) should you configure some interfaces to be Level 2-only to prevent wasting bandwidth by sending out unused Level 1 hello packets. Remember that on point-to-point interfaces, the Level 1 and Level 2 hello packets are in the same packet.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure a Level 1 adjacency with its neighbor on tenGigE interface 0/2/0/0 and Level 2 adjacencies with all Level 2-capable routers on tenGigE interface 0/5/0/2:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# is-type level-1-2
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/0
RP/0/RP0/CPU0:router(config-isis-if)# circuit-type level-1
RP/0/RP0/CPU0:router(config-isis-if)# exit
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/5/0/2
RP/0/RP0/CPU0:router(config-isis-if)# circuit-type level-2-only
```
In this example, only Level 2 adjacencies are established because the **is-type** command is configured:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# is-type level-2-only
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/0
RP/0/RP0/CPU0:router(config-isis-if)# circuit-type level-1-2
```
clear isis process

To clear the link-state packet (LSP) database and adjacency database sessions for an Intermediate System-to-Intermediate System (IS-IS) instance or all IS-IS instances, use the `clear isis process` command in XR EXEC mode.

```
clear isis [instance instance-id] process
```

**Syntax Description**

- `instance instance-id` (Optional) Specifies IS-IS sessions for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `clear isis process` command without any keyword to clear all the IS-IS instances. Add the `instance instance-id` keyword and argument to clear the specified IS-IS instance.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows the IS-IS LSP database and adjacency sessions being cleared for instance 1:

```
RP/0/RP0/CPU0:router# clear isis instance 1 process
```
To clear the Intermediate System-to-Intermediate System (IS-IS) routes in a topology, use the **clear isis route** command in XR EXEC mode.

**clear isis route**

**Syntax Description**

```
instance instance-id (Optional) Specifies IS-IS sessions for the specified IS-IS instance only.
  • The instance-id argument is the instance identifier (alphanumeric) defined by the router isis command.

afi-all Specifies IP Version 4 (IPv4) and IP Version 6 (IPv6) address prefixes.
ipv4 Specifies IPv4 address prefixes.
ipv6 Specifies IPv6 address prefixes.
unicast Specifies unicast address prefixes.
safi-all Specifies all secondary address prefixes.

(topology topo-name (Optional) Specifies topology table information and name of the topology table.
```

**Command Default**

No default behavior or value

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **clear isis route** command to clear the routes from the specified topology or all routes in all topologies if no topology is specified.

**Task ID**

```
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>execute</td>
</tr>
<tr>
<td>rib</td>
<td>read, write</td>
</tr>
<tr>
<td>basic-services</td>
<td>read, write</td>
</tr>
</tbody>
</table>
```

**Examples**

The following example shows how to clear the routes with IPv4 unicast address prefixes:

```
RP/0/RP0/CPU0:router# clear isis ipv4 unicast route
```
clear isis statistics

To clear the Intermediate System-to-Intermediate System (IS-IS) statistics, use the `clear isis statistics` command in XR EXEC mode.

```
clear isis [instance instance-id] statistics [type interface-path-id]
```

**Syntax Description**

- **instance instance-id** (Optional) Clears IS-IS sessions for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.
- **type** Interface type. For more information, use the question mark (?) online help function.
- **interface-path-id** Physical interface or virtual interface.

**Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

Release 6.0 This command was introduced.

**Usage Guidelines**

Use the `clear isis statistics` command to clear the information displayed by the `show isis statistics` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>execute</td>
</tr>
<tr>
<td>rib</td>
<td>read, write</td>
</tr>
<tr>
<td>basic-services</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows the IS-IS statistics for a specified interface being cleared:

```
RP/0/RP0/CPU0:router# clear isis instance 23 statistics
```
**csnp-interval**

To configure the interval at which periodic complete sequence number PDU (CSNP) packets are sent on broadcast interfaces, use the `csnp-interval` command in interface configuration mode. To restore the default value, use the `no` form of this command.

```
csnp-interval seconds [level {1|2}]
nocsnp-interval seconds [level {1|2}]
```

**Syntax Description**

`seconds` Interval (in seconds) of time between transmission of CSNPs on multiaccess networks. This interval applies only for the designated router. Range is 0 to 65535 seconds.

`level {1 | 2}` (Optional) Specifies the interval of time between transmission of CSNPs for Level 1 or Level 2 independently.

**Command Default**

`seconds`: 10 seconds

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

Interface configuration

**Command History**

Release Modification

Release 6.0 This command was introduced.

**Usage Guidelines**

The `csnp-interval` command applies only to the designated router (DR) for a specified interface. Only DRs send CSNP packets to maintain database synchronization. The CSNP interval can be configured independently for Level 1 and Level 2.

Use of the `csnp-interval` command on point-to-point subinterfaces makes sense only in combination with the IS-IS mesh-group feature.

**Examples**

The following example shows how to set the CSNP interval for Level 1 to 30 seconds:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/0/2/0
RP/0/RP0/CPU0:router(config-isis-if)# csnp-interval 30 level 1
```
default-information originate (IS-IS)

To generate a default route into an Intermediate System-to-Intermediate System (IS-IS) routing domain, use the `default-information originate` command in address family configuration mode. To remove the `default-information originate` command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
default-information originate [{external|route-policy route-policy-name}]  
no default-information originate [{external|route-policy route-policy-name}]  
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>external</code></td>
<td>(Optional) Enables default route to be originated as an external route.</td>
</tr>
<tr>
<td><code>route-policy</code></td>
<td>(Optional) Defines the conditions for the default route.</td>
</tr>
<tr>
<td><code>route-policy-name</code></td>
<td>(Optional) Name for the route policy.</td>
</tr>
</tbody>
</table>

**Command Default**

A default route is not generated into an IS-IS routing domain.

**Command Modes**

Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If a router configured with the `default-information originate` command has a route to 0.0.0.0 in the routing table, IS-IS originates an advertisement for 0.0.0.0 in its link-state packets (LSPs).

Without a route policy, the default is advertised only in Level 2 LSPs. For Level 1 routing, there is another process to find the default route, which is to look for the closest Level 1 and Level 2 router. The closest Level 1 and Level 2 router can be found by looking at the attached-bit (ATT) in Level 1 LSPs.

A route policy can be used for two purposes:

- To make the router generate the default route in its Level 1 LSPs.
- To advertise 0.0.0.0/0 conditionally.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to generate a default external route into an IS-IS domain:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
```
default-information originate

RP/0/RP0/CPU0:router(config-isis-af)# default-information originate
disable (IS-IS)

To disable the Intermediate System-to-Intermediate System (IS-IS) topology on a specified interface, use the disable command in interface address family configuration mode. To remove this function, use the no form of this command.

disable
no disable

Command Default
IS-IS protocol is enabled.

Command Modes
Interface address family configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines
No specific guidelines impact the use of this command.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples

The following example shows how to disable the IS-IS protocol for IPv4 unicast on tenGigE interface 0/1/0/1:

RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# disable
distance (IS-IS)

To define the administrative distance assigned to routes discovered by the Intermediate System-to-Intermediate System (IS-IS) protocol, use the `distance` command in address family configuration mode. To remove the `distance` command from the configuration file and restore the system to its default condition in which the software removes a distance definition, use the `no` form of this command.

```
distance weight [{prefix maskprefix/length [{prefix-list-name}]}]
no distance [{weight}] [{prefix maskprefix/length[\{prefix-list-name\}]]}
```

**Syntax Description**

- `weight`: Administrative distance to be assigned to IS-IS routes. Range is 1 to 255.
- `prefix`: (Optional) The `prefix` argument specifies the IP address in four-part, dotted-decimal notation.
- `mask`: (Optional) IP address mask.
- `/length`: (Optional) The length of the IP prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash must precede the decimal value. Range is 0 to 32 for IPv4 addresses and 0 to 128 for IPv6 addresses.
- `prefix-list-name`: (Optional) List of routes to which administrative distance applies.

**Command Default**

`weight : 115`

**Command Modes**

Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

An administrative distance is an integer from 1 to 255. In general, the higher the value, the lower the trust rating. An administrative distance of 255 means that the routing information source cannot be trusted at all and should be ignored. Weight values are subjective; no quantitative method exists for choosing weight values.

Use the `distance` command to configure the administrative distances applied to IS-IS routes when they are inserted into the Routing Information Base (RIB), and influence the likelihood of these routes being preferred over routes to the same destination addresses discovered by other protocols.

The `address/prefix-length` argument defines to which source router the distance applies. In other words, each IS-IS route is advertised by another router, and that router advertises an address that identifies it. This source address is displayed in the output of the `show isis route detail` command.

The `distance` command applies to the routes advertised by routers whose address matches the specified prefix. The `prefix-list-name` argument can then be used to refine this further so that the `distance` command affects only specific routes.
## Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

## Examples

In the following example, a distance of 10 is assigned to all routes to 2.0.0.0/8 and 3.0.0.0/8 (or more specific prefixes) that are advertised by routers whose ID is contained in 1.0.0.0/8. A distance of 80 is assigned to all other routes.

```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list target_routes
RP/0/RP0/CPU0:router(config-ipv4_pfx)# permit 2.0.0.0/8
RP/0/RP0/CPU0:router(config-ipv4_pfx)# permit 3.0.0.0/8
RP/0/RP0/CPU0:router(config-ipv4_pfx)# deny 0.0.0.0/0
RP/0/RP0/CPU0:router(config-ipv4_pfx)# exit
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# distance 10 1.0.0.0/8 target_routes
RP/0/RP0/CPU0:router(config-isis-af)# distance 80
```
hello-interval (IS-IS)

To specify the length of time between consecutive hello packets sent by the Intermediate System-to-Intermediate System (IS-IS) protocol software, use the **hello-interval** command in interface configuration mode. To restore the default value, use the **no** form of this command.

```
hello-interval seconds [level {1|2}]
no hello-interval [seconds] [level {1|2}]
```

**Syntax Description**

- **seconds**: Integer value (in seconds) for the length of time between consecutive hello packets. By default, a value three times the hello interval *seconds* is advertised as the **hold time** in the hello packets sent. (That multiplier of three can be changed by using the **hello-multiplier** command.) With smaller hello intervals, topological changes are detected more quickly, but there is more routing traffic. Range is 1 to 65535 seconds.

- **level**: {1 | 2} (Optional) Specifies the hello interval for Level 1 and Level 2 independently. For broadcast interfaces only.

**Command Default**

- **seconds**: 10 seconds

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The hello interval can be configured independently for Level 1 and Level 2, except on serial point-to-point interfaces. (Because only a single type of hello packet is sent on serial links, it is independent of Level 1 or Level 2.) Configuring Level 1 and Level 2 independently is used on LAN interfaces.

**Note**

- A shorter hello interval gives quicker convergence, but increases bandwidth and CPU usage. It might also add to instability in the network.
- A slower hello interval saves bandwidth and CPU. Especially when used in combination with a higher hello multiplier, this strategy may increase overall network stability.
- For point-to-point links, IS-IS sends only a single hello for Level 1 and Level 2, making the **level** keyword meaningless on point-to-point links. To modify hello parameters for a point-to-point interface, omit the **level** keyword.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to configure tenGigE 0/6/0/0 to advertise hello packets every 5 seconds for Level 1 topology routes. This situation causes more traffic than configuring a longer interval, but topological changes are detected more quickly.

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/6/0/0
RP/0/RP0/CPU0:router(config-isis-if)# hello-interval 5 level 1
```
hello-multiplier

To specify the number of Intermediate System-to-Intermediate System (IS-IS) hello packets a neighbor must miss before the router should declare the adjacency as down, use the `hello-multiplier` command in interface configuration mode. To restore the default value, use the `no` form of this command.

```
hello-multiplier multiplier [level {1/2}]
no hello-multiplier [multiplier] [level {1/2}]
```

**Syntax Description**

- `multiplier`:
  - Advertised hold time in IS-IS hello packets is set to the hello multiplier times the hello interval. Range is 3 to 1000. Neighbors declare an adjacency to this down router after not having received any IS-IS hello packets during the advertised hold time. The hold time (and thus the hello multiplier and the hello interval) can be set on an individual interface basis, and can be different between different networking devices in one area.
  - Using a smaller hello multiplier gives faster convergence, but can result in more routing instability. Increase the hello multiplier to a larger value to help network stability when needed. Never configure a hello multiplier to a value lower than the default value of 3.

- `level {1 | 2}`:
  - (Optional) Specifies the hello multiplier independently for Level 1 or Level 2 adjacencies.

**Command Default**

```
multiplier : 3
```

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

Interface configuration

**Command History**

```
Release   Modification
---------  ----------------
Release 6.0  This command was introduced.
```

**Usage Guidelines**

The “holding time” carried in an IS-IS hello packet determines how long a neighbor waits for another hello packet before declaring the neighbor to be down. This time determines how quickly a failed link or neighbor is detected so that routes can be recalculated.

Use the `hello-multiplier` command in circumstances where hello packets are lost frequently and IS-IS adjacencies are failing unnecessarily. You can raise the hello multiplier and lower the hello interval (hello-interval (IS-IS), on page 19 command) correspondingly to make the hello protocol more reliable without increasing the time required to detect a link failure.

Use the `hello-multiplier` command in circumstances where hello packets are lost frequently and IS-IS adjacencies are failing unnecessarily. You can raise the hello multiplier and lower the hello interval (hello-interval (IS-IS), on page 19 command) correspondingly to make the hello protocol more reliable without increasing the time required to detect a link failure.

On point-to-point links, there is only one hello for both Level 1 and Level 2. Separate Level 1 and Level 2 hello packets are also sent over nonbroadcast multiaccess (NBMA) networks in multipoint mode, such as X.25, Frame Relay, and ATM.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how the network administrator wants to increase network stability by making sure an adjacency goes down only when many (ten) hello packets are missed. The total time to detect link failure is 60 seconds. This strategy ensures that the network remains stable, even when the link is fully congested.

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-isis-if)# hello-interval 6
RP/0/RP0/CPU0:router(config-isis-if)# hello-multiplier 10
```
hello-padding

To configure padding on Intermediate System-to-Intermediate System (IS-IS) hello protocol data units (IIH PDUs) for all IS-IS interfaces on the router, use the **hello-padding** command in interface configuration mode. To suppress padding, use the **no** form of this command.

```
hello-padding  {disable|sometimes}  [level  {1|2}]  
no  hello-padding  {disable|sometimes}  [level  {1|2}]  
```

**Syntax Description**

- **disable** Suppresses hello padding.
- **sometimes** Enables hello padding during adjacency formation only.
- **level**  
  - `{ 1 | 2 }` (Optional) Specifies hello padding for Level 1 or Level 2 independently.

**Command Default**

Hello padding is enabled.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You might want to suppress hello padding to conserve network resources. The lower the circuit speed, the higher the percentage of padding overhead. Before suppressing the hello padding, you should know your physical and data link layer configurations and have control over them, and also know your router configuration at the network layer.

For point-to-point links, IS-IS sends only a single hello for Level 1 and Level 2, making the **level** keyword meaningless on point-to-point links. To modify hello parameters for a point-to-point interface, omit the **level** keyword.

**Task ID**

**Task ID**

- **isis** read, write

**Examples**

The following example shows how to suppress IS-IS hello padding over local area network (LAN) circuits for interface tenGigE 0/2/0/1:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-isis-if)# hello-padding disable
```
**hello-password**

To configure the authentication password for an Intermediate System-to-Intermediate System (IS-IS) interface, use the **hello-password** command in interface configuration mode. To disable authentication, use the **no** form of this command.

```
hello-password [ { hmac-md5 | text } ] [ { clear | encrypted } ] password [ level { 1 | 2 } ] [ send-only ]
no hello-password [ { hmac-md5 | text } ] [ { clear | encrypted } ] password [ level { 1 | 2 } ] [ send-only ]
```

**Syntax Description**
- `hmac-md5` (Optional) Specifies that the password use HMAC-MD5 authentication.
- `text` (Optional) Specifies that the password use clear text password authentication.
- `clear` (Optional) Specifies that the password be unencrypted.
- `encrypted` (Optional) Specifies that the password be encrypted using a two-way algorithm.
- `password` Authentication password you assign for an interface.
- `level { 1 | 2 }` (Optional) Specifies whether the password is for a Level 1 or a Level 2 protocol data unit (PDU).
- `send-only` (Optional) Specifies that the password applies only to protocol data units (PDUs) that are being sent and does not apply to PDUs that are being received.

**Command Default**
Both Level 1 and Level 2 are configured if no level is specified.

- `password`: encrypted text

**Command Modes**
Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
When a `text` password is configured, it is exchanged as clear text. Therefore, the **hello-password** command provides limited security.

When an `hmac-md5` password is configured, the password is never sent over the network and is instead used to calculate a cryptographic checksum to ensure the integrity of the exchanged data.

For point-to-point links, IS-IS sends only a single hello for Level 1 and Level 2, making the `level` keyword meaningless on point-to-point links. To modify hello parameters for a point-to-point interface, omit the `level` keyword.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to configure a password with HMAC-MD5 authentication for hello packets running on tenGigE 0/2/0/3 interface:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/3
RP/0/RP0/CPU0:router(config-isis-if)# hello-password hmac-md5 clear mypassword
```
hello-password keychain

To configure the authentication password keychain for an Intermediate System-to-Intermediate System (IS-IS) interface, use the `hello-password keychain` command in interface configuration mode. To disable the authentication password keychain, use the `no` form of this command.

```
hello-password keychain keychain-name [level {1|2}] [send-only]
no hello-password keychain keychain-name [level {1|2}] [send-only]
```

**Syntax Description**

- **keychain**  Keyword that specifies the keychain to be configured. An authentication password keychain is a sequence of keys that are collectively managed and used for authenticating a peer-to-peer group.
- **keychain-name**  Specifies the name of the keychain.
- **level {1|2}**  (Optional) Specifies whether the keychain is for a Level 1 or a Level 2 protocol data unit (PDU).
- **send-only**  (Optional) Specifies that the keychain applies only to protocol data units (PDUs) that are being sent and does not apply to PDUs that are being received.

**Command Default**

Both Level 1 and Level 2 are configured if no level is specified.

**password**: encrypted text

**Command Modes**

Interface configuration

**Command History**

- **Release 6.0**  This command was introduced.

**Usage Guidelines**

Specify a keychain to enable keychain authentication between two IS-IS peers. Use the `keychain keychain-name` keyword and argument to implement hitless key rollover for authentication.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure a password keychain for level 1, send only authentication on a tenGigE:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/1/0/0
RP/0/RP0/CPU0:router(config-isis-if)# hello-password keychain mykeychain level 1 send-only
```
hello-password accept

To configure an additional authentication password for an Intermediate System-to-Intermediate System (IS-IS) interface, use the **hello-password accept** command in interface configuration mode. To disable authentication, use the **no** form of this command.

```
hello-password accept {clear|encrypted} password [level {1|2}]
no hello-password accept {clear|encrypted} password [level {1|2}]
```

**Syntax Description**
- **clear** Specifies that the password be unencrypted.
- **encrypted** Specifies that the password be encrypted using a two-way algorithm.
- **password** Authentication password you assign.
- **level { 1 | 2 }** (Optional) Specifies the password for Level 1 or Level 2 independently.

**Command Default**
Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**
Interface configuration

**Command History**
```
Release   Modification
----------  ------------
6.0        This command was introduced.
```

**Usage Guidelines**
Use the **hello-password accept** command to add an additional password for an IS-IS interface. An authentication password must be configured using the **hello-password** command before an accept password can be configured for the corresponding level.

**Task ID**
```
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>
```

**Examples**
The following example shows how to configure a password:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/3
RP/0/RP0/CPU0:router(config-isis)# hello-password accept encrypted 111D1C1603
```
hostname dynamic disable

To disable Intermediate System-to-Intermediate System (IS-IS) routing protocol dynamic hostname mapping, use the `hostname dynamic` command in XR Config mode. To remove the specified command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
hostname dynamic disable
no hostname dynamic disable
```

**Syntax Description**

<table>
<thead>
<tr>
<th>disable</th>
<th>Disables dynamic host naming.</th>
</tr>
</thead>
</table>

**Command Default**

Router names are dynamically mapped to system IDs.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

In an IS-IS routing domain, each router is represented by a 6-byte hexadecimal system ID. When network administrators maintain and troubleshoot networking devices, they must know the router name and corresponding system ID.

Link-state packets (LSPs) include the dynamic hostname in the type, length, and value (TLV) which carries the mapping information across the entire domain. Every router in the network, upon receiving the TLV from an LSP, tries to install it in a mapping table. The router then uses the mapping table when it wants to convert a system ID to a router name.

To display the entries in the mapping tables, use the `show isis hostname` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to disable dynamic mapping of hostnames to system IDs:

```
RP/0/RP0/CPU0:router(config)# router isis lsp
RP/0/RP0/CPU0:router(config-isis)# hostname dynamic disable
```
ignore-lsp-errors

To override the default setting of a router to ignore Intermediate System-to-Intermediate System (IS-IS) link-state packets (LSPs) that are received with internal checksum errors, use the `ignore-lsp-errors disable` command in XR Config mode. To enable ignoring IS-IS LSP errors, use the `no` form of this command.

```
ignore-lsp-errors disable
no ignore-lsp-errors disable
```

**Syntax Description**
- **disable** Disables the functionality of the command.

**Command Default**
The system ignores corrupt LSPs.

**Command Modes**
XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**
The IS-IS protocol definition requires that a received LSP with an incorrect data-link checksum be purged by the receiver, which causes the initiator of the packet to regenerate it. However, if a network has a link that causes data corruption and at the same time is delivering LSPs with correct data-link checksums, a continuous cycle of purging and regenerating large numbers of packets can occur. Because this situation could render the network nonfunctional, use this command to ignore these LSPs rather than purge the packets.

The receiving network devices use link-state packets to maintain their routing tables.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to instruct the router to purge LSPs that cause the initiator to regenerate LSPs:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# ignore-lsp-errors disable
```
instance-id

To allow a router to share one or more circuits among multiple Intermediate System to Intermediate System (IS-IS) routing protocol instances, use the **instance-id** command in router configuration mode.

```plaintext
instance-id identifier
```

**Syntax Description**

- **identifier**
  - Range is 1-65535.

**Command Default**

Disabled

**Command Modes**

Router configuration

**Command History**

- **Release 6.1.x**
  - This command was introduced.

**Task ID**

- **Task ID**
  - isis
  - read, write

**Examples**

The following example shows how to configure multiple instances on a single router:

```plaintext
RP/0/RP0/CPU0:router(config)# router isis ring
RP/0/RP0/CPU0:router(config-isis)# instance-id 1
RP/0/RP0/CPU0:router(config-isis)# exit
RP/0/RP0/CPU0:router(config)# router isis 1
RP/0/RP0/CPU0:router(config-isis)# instance-id 6
RP/0/RP0/CPU0:router(config-isis)#
```
interface (IS-IS)

To configure the Intermediate System-to-Intermediate System (IS-IS) protocol on an interface, use the `interface` command in XR Config mode. To disable IS-IS routing for interfaces, use the `no` form of this command.

```
interface type interface-path-id
no interface type interface-path-id
```

**Syntax Description**

- **type** Interface type. For more information, use the question mark (`?`) online help function.
- **interface-path-id** Physical interface or virtual interface.

**Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (`?`) online help function.

**Command Default**

No interfaces are specified.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

An address family must be established on the IS-IS interface before the interface is enabled for IS-IS protocol operation.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to enable an IS-IS multitopology configuration for IPv4 on tenGigE interface 0/3/0/0:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# net 49.0000.0000.0001.00
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# metric-style wide level 1
! RP/0/RP0/CPU0:router(config)# interface tenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 2001::1/64
```
ispf

To configure the incremental shortest path first (iSPF) algorithm to calculate network topology, use the `ispf` command in address family configuration mode. To disable this algorithm function, use the `no` form of this command.

```
ispf [level {1|2}]
no ispf [level {1|2}]
```

| Syntax Description | level { 1 | 2 } (Optional) Configures the iSPF algorithm for Level 1 or Level 2 independently. |
|--------------------|-------------------------------------------------------------------------------------------------|
| Command Default    | The iSPF algorithm is not configured.                                                           |
| Command Modes      | Address family configuration                                                                    |
| Command History    | Release 6.0  This command was introduced.                                                      |
| Usage Guidelines   | The iSPF algorithm may be used to reduce the processor load when IS-IS needs to recalculate its topology after minor changes. |

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Examples</th>
</tr>
</thead>
</table>

The following example shows how to configure iSPF for the IPv4 unicast topology at Level 1:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# ispf level 1
```
To configure the routing level for an Intermediate System-to-Intermediate System (IS-IS) area, use the **is-type** command in XR Config mode. To set the routing level to the default level, use the **no** form of this command.

```plaintext
is-type  {level-1|level-1-2|level-2-only}
no  is-type  [{level-1|level-1-2|level-2-only}]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>level-1</strong></td>
<td>Specifies that the router perform only Level 1 (intra-area) routing. This router learns only about destinations inside its area. Level 2 (interarea) routing is performed by the closest Level 1-2 router.</td>
</tr>
<tr>
<td><strong>level-1-2</strong></td>
<td>Specifies that the router perform both Level 1 and Level 2 routing.</td>
</tr>
<tr>
<td><strong>level-2-only</strong></td>
<td>Specifies that the routing process acts as a Level 2 (interarea) router only. This router is part of the backbone, and does not communicate with Level 1-only routers in its own area.</td>
</tr>
</tbody>
</table>

### Command Default

Both Level 1 and Level 2 are configured if no level is specified.

### Command Modes

XR Config mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

When the router is configured with Level 1 routing only, this router learns about destinations only inside its area. Level 2 (interarea) routing is performed by the closest Level 1-2 router.

When the router is configured with Level 2 routing only, this router is part of the backbone, and does not communicate with Level 1 routers in its own area.

The router has one link-state packet database (LSDB) for destinations inside the area (Level 1 routing) and runs a shortest path first (SPF) calculation to discover the area topology. It also has another LSDB with link-state packets (LSPs) of all other backbone (Level 2) routers, and runs another SPF calculation to discover the topology of the backbone and the existence of all other areas.

We highly recommend that you configure the type of an IS-IS routing process to establish the proper level of adjacencies. If there is only one area in the network, there is no need to run both Level 1 and Level 2 routing algorithms.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to specify that the router is part of the backbone and that it does not communicate with Level 1-only routers:
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# is-type level-2-only
log pdu drops

To log Intermediate System-to-Intermediate System (IS-IS) protocol data units (PDUs) that are dropped, use the `log pdu drops` command in XR Config mode. To disable this function, use the `no` form of this command.

```
log pdu drops

no log pdu drops
```

**Command Default**  
PDU logging is disabled.

**Command Modes**  
XR Config mode

**Command History**  

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**  
Use the `log pdu drops` command to monitor a network when IS-IS PDUs are suspected of being dropped. The reason for the PDU being dropped and current PDU drop statistics are recorded.

The following are examples of PDU logging output:

```
%ISIS-4-ERR_IIH_INPUT_Q_OVERFLOW: IIH input queue overflow: 86 total drops; 19 IIH drops, 44 LSP drops, 23 SNP drops
%ISIS-4-ERR_LSP_INPUT_Q_OVERFLOW: LSP input queue overflow: 17 total drops; 9 IIH drops, 3 LSP drops, 5 SNP drops
```

**Examples**  
The following example shows how to enable PDU logging:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# log pdu drops
```
**Isp fast-flood threshold**

To configure the link-state packet (LSP) fast-flood threshold, use the `lsp fast-flood threshold` command in interface configuration mode. To restore the default value, use the `no` form of this command.

```
lsp fast-flood threshold lsp-number [level {1|2}]
no lsp fast-flood threshold [lsp-number] [level {1|2}]
```

**Syntax Description**
- `lsp-number` (Optional) Number of LSPs to send back to back. Range is 1 to 4294967295.
- `level {1 | 2}` (Optional) Specifies the LSP threshold for Level 1 or Level 2 independently.

**Command Default**
10 LSPs are allowed in a back-to-back window

**Command Modes**
Interface configuration

**Command History**
- Release 6.0  This command was introduced.

**Usage Guidelines**
Use the `lsp fast-flood threshold` command to accelerate convergence of LSP database. LSPs are sent back-to-back over an interface up to the specified limit. Past the limit, LSPs are sent out in the next batch window as determined by LSP pacing interval.

Duration of back-to-back window = LSP interval * LSP fast-flood threshold limit.

**Examples**
The following example shows how to configure the LSP threshold:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-isis-if)# lsp fast-flood threshold 234 level 1
```
Isp-gen-interval

To customize IS-IS throttling of link-state packet (LSP) generation, use the `lsp-gen-interval` command in XR Config mode. To restore the default value, use the `no` form of this command.

```
Isp-gen-interval [initial-wait initial] [secondary-wait secondary] [maximum-wait maximum] [level {1|2}]
no lsp-gen-interval [initial-wait initial] [secondary-wait secondary] [maximum-wait maximum] [level {1|2}]
```

### Syntax Description

- **initial-wait initial**: Specifies the initial LSP generation delay (in milliseconds). Range is 0 to 120000 milliseconds.
- **secondary-wait secondary**: Specifies the hold time between the first and second LSP generation (in milliseconds). Range is 1 to 120000 milliseconds.
- **maximum-wait maximum**: Specifies the maximum interval (in milliseconds) between two consecutive occurrences of an LSP being generated. Range is 1 to 120000 milliseconds.
- **level {1 | 2}**: (Optional) Specifies the LSP time interval for Level 1 or Level 2 independently.

### Command Default

- **initial-wait initial**: 50 milliseconds
- **secondary-wait secondary**: 200 milliseconds
- **maximum-wait maximum**: 5000 milliseconds

### Command Modes

XR Config mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

### Usage Guidelines

During prolonged periods of network instability, repeated recalculation of LSPs can cause increased CPU load on the local router. Further, the flooding of these recalculated LSPs to the other Intermediate Systems in the network causes increased traffic and can result in other routers having to spend more time running route calculations.

Use the `lsp-gen-interval` command to reduce the rate of LSP generation during periods of instability in the network. This command can help to reduce CPU load on the router and to reduce the number of LSP transmissions to its IS-IS neighbors.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to set the maximum interval between two consecutive occurrences of an LSP to 15 milliseconds and the initial LSP generation delta to 5 milliseconds:

```
RP/0/RP0/CPU0:router(config)# router isisisp
RP/0/RP0/CPU0:router(config-isis)# lsp-gen-interval maximum-wait 15 initial-wait 5
```
Isp-interval

To configure the amount of time between consecutive link-state packets (LSPs) sent on an Intermediate System-to-Intermediate System (IS-IS) interface, use the `lsp-interval` command in interface configuration mode. To restore the default value, use the `no` form of this command.

```
Isp-interval milliseconds [level {1|2}]
no lsp-interval [milliseconds] [level {1|2}]
```

**Syntax Description**

- `milliseconds` Time delay (in milliseconds) between successive LSPs. Range is 1 to 4294967295.
- `level {1 | 2}` (Optional) Configures the LSP time delay for Level 1 or Level 2 independently.

**Command Default**

- `milliseconds`: 33 milliseconds

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to cause the system to send LSPs every 100 milliseconds (10 packets per second) on Level 1 and Level 2:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-isis-if)# lsp-interval 100
```
**lsp-mtu**

To set the maximum transmission unit (MTU) size of Intermediate System-to-Intermediate System (IS-IS) link-state packets (LSPs), use the **lsp-mtu** command in XR Config mode. To restore the default, use the **no** form of this command.

```
lsp-mtu bytes [level {1|2}]  
no lsp-mtu [bytes] [level {1|2}]  
```

**Syntax Description**

- **bytes**: Maximum packet size in bytes. The number of bytes must be less than or equal to the smallest MTU of any link in the network. Range is 128 to 4352 bytes.
- **level {1 | 2}**: (Optional) Specifies routing Level 1 or Level 2 independently.

**Command Default**

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Under normal conditions, the default MTU size should be sufficient. However, if the MTU size of a link is less than 1500 bytes, the LSP MTU size must be lowered accordingly on each router in the network. If this action is not taken, routing becomes unpredictable.

This guideline applies to all Cisco networking devices in a network. If any link in the network has a reduced MTU size, all devices must be changed, not just the devices directly connected to the link.

**Note**

Do not set the **lsp-mtu** command (network layer) to a value greater than the link MTU size that is set with the **mtu** command (physical layer).

To be certain about a link MTU size, use the **show isis interface, on page 105** command to display the value.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the MTU size to 1300 bytes:

```
RP/0/RP0/CPU0:router(config)# router isis lsp
RP/0/RP0/CPU0:router(config-isis)# lsp-mtu 1300
```
**lsp-password**

To configure the link-state packet (LSP) authentication password, use the `lsp-password` command in XR Config mode. To remove the `lsp-password` command from the configuration file and disable link-state packet authentication, use the `no` form of this command.

```
lsp-password [{{hmac-md5|text}} [{{clear|encrypted}}] password|keychain keychain-name] [level {1|2}] [send-only] [snp send-only] | [enable-poi]
no lsp-password [{{hmac-md5|text}} [{{clear|encrypted}}] password|keychain keychain-name] [level {1|2}] [send-only] [snp send-only] | [enable-poi]
```

**Syntax Description**

- **hmac-md5**: Specifies that the password uses HMAC-MD5 authentication.
- **text**: Specifies that the password uses clear text password authentication.
- **clear**: Specifies that the password be unencrypted.
- **encrypted**: Specifies that the password be encrypted using a two-way algorithm.
- **password**: Authentication password you assign.
- **keychain**: (Optional) Specifies a keychain.
- **keychain-name**: Name of the keychain.
- **level {1 | 2}**: (Optional) Specifies the password for Level 1 or Level 2 independently.
- **send-only**: (Optional) Adds passwords to LSP and sequence number protocol (SNP) data units when they are sent. Does not check for authentication in received LSPs or sequence number PDUs (SNPs).
- **snp send-only**: (Optional) Adds passwords to SNP data units when they are sent. Does not check for authentication in received SNPs. This option is available when the `text` keyword is specified.
- **enable-poi**: The `enable-poi` keyword inserts the purge originator identification (POI), if you are using cryptographic authentication. If you are not using cryptographic authentication, then the POI is inserted by default.

**Command Default**

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>Release 6.5.1</td>
<td>The <code>enable-poi</code> keyword is added.</td>
</tr>
</tbody>
</table>
**Usage Guidelines**

When a **text** password is configured, it is exchanged as clear text. Therefore, the `lsp-password` command provides limited security.

When an **HMAC-MD5** password is configured, the password is never sent over the network and is instead used to calculate a cryptographic checksum to ensure the integrity of the exchanged data.

The recommended password configuration is that both incoming and outgoing SNPs be authenticated.

---

**Note**

To disable SNP password checking, the `snp send-only` keywords must be specified in the `lsp-password` command.

To configure an additional password, use the `lsp-password accept` command.

Specify a key chain to enable key chain authentication between two IS-IS peers. Use the `keychain keychain-name` keyword and argument to implement hitless key rollover for authentication.

---

**Examples**

The following example shows how to configure separate Level 1 and Level 2 LSP and SNP passwords, one with HMAC-MD5 authentication and encryption and one with clear text password authentication and no encryption:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# lsp-password hmac-md5 clear password1 level 1
RP/0/RP0/CPU0:router(config-isis)# lsp-password text clear password2 level 2
```
lsp-password accept

To configure an additional link-state packet (LSP) authentication password, use the `lsp-password accept` command in XR Config mode. To remove the `lsp-password accept` command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
lsp-password accept {clear|encrypted} password [level {1|2}]
no lsp-password accept [{clear|encrypted} password [level {1|2}]]
```

**Syntax Description**
- **clear**: Specifies that the password be unencrypted.
- **encrypted**: Specifies that the password be encrypted using a two-way algorithm.
- **password**: Authentication password you assign.
- **level {1 | 2}**: (Optional) Specifies the password for Level 1 or Level 2 independently.

**Command Default**
Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**
XR Config mode

**Command History**
- **Release 6.0**: This command was introduced.

**Usage Guidelines**
The `lsp-password accept` command adds an additional password for use when the system validates incoming LSPs and sequence number PDUs (SNPs). An LSP password must be configured using the `lsp-password` command before an accept password can be configured for the corresponding level.

**Task ID**
- **Task ID**
  - ** Operations**
    - **isis** read, write

**Examples**
The following example shows how to configure an accept Level 1 LSP and SNP password:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# lsp-password encrypted password1 level 1
```
lsp-refresh-interval

To set the time between regeneration of link-state packets (LSPs) that contain different sequence numbers, use the **lsp-refresh-interval** command in XR Config mode. To restore the default refresh interval, use the **no** form of this command.

```
  Command History
  Release  Modification
  6.0     This command was introduced.
```

---

**Syntax Description**

- **seconds** (Refresh interval (in seconds). Range is 1 to 65535 seconds.)
- **level** (Optional) Specifies routing Level 1 or Level 2 independently.

---

**Examples**

```
  RP/0/RP0/CPU0:router(config)# router isis lsp
  RP/0/RP0/CPU0:router(config-isis)# lsp-refresh-interval 10800
```
maximum-paths (IS-IS)

To configure the maximum number of parallel routes that an IP routing protocol will install in the routing table, use the `maximum-paths` command in address family configuration mode. To remove the `maximum-paths` command from the configuration file and restore the system to its default condition with respect to the routing protocol, use the `no` form of this command.

```
maximum-paths maximum
no maximum-paths
```

**Syntax Description**

- `maximum` Maximum number of parallel routes that IS-IS can install in a routing table. Range is 1 to 32.

**Command Default**

1 to 8 routes

**Command Modes**

Address family configuration

**Command History**

- **Release**
  - 6.0  This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to allow a maximum of 16 paths to a destination:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# maximum-paths 16
```
maximum-redistributed-prefixes (IS-IS)

To specify an upper limit on the number of redistributed prefixes (subject to summarization) that the Intermediate System-to-Intermediate System (IS-IS) protocol advertises, use the `maximum-redistributed-prefixes` command in address family mode. To disable this feature, use the `no` form of this command.

```
maximum-redistributed-prefixes maximum [level {1|2}]
no maximum-redistributed-prefixes [maximum [level {1|2}]]
```

**Syntax Description**
- `maximum`: Maximum number of redistributed prefixes advertised. Range is 1 to 28000.
- `level {1 | 2}`: (Optional) Specifies maximum prefixes for Level 1 or Level 2.

**Command Default**
- `maximum`: 10000
- `level`: 1-2

**Command Modes**
- Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `maximum-redistributed-prefixes` command to prevent a misconfiguration from resulting in redistribution of excess prefixes. If IS-IS encounters more than the maximum number of prefixes, it sets a bi-state alarm. If the number of to-be-redistributed prefixes drops back to the maximum or lower—either through reconfiguration or a change in the redistribution source—IS-IS clears the alarm.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to specify the number of redistributed prefixes at 5000 for Level 2:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# maximum-redistributed-prefixes 5000 level 2
```
max-lsp-lifetime

To set the maximum time that link-state packets (LSPs) persist without being refreshed, use the `max-lsp-lifetime` command in XR Config mode. To restore the default time, use the `no` form of this command.

```
max-lsp-lifetime  seconds  [level  {1|2}]
no   max-lsp-lifetime  [seconds  [level  {1|2}]]
```

**Syntax Description**

- `seconds` Lifetime (in seconds) of the LSP. Range from 1 to 65535 seconds.
- `level {1 | 2}` (Optional) Specifies routing Level 1 or Level 2 independently.

**Command Default**

- `seconds` : 1200 seconds (20 minutes)
- Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You might need to adjust the maximum LSP lifetime if you change the LSP refresh interval with the `lsp-refresh-interval` command. The maximum LSP lifetime must be greater than the LSP refresh interval.

**Task ID**

```
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read,</td>
</tr>
<tr>
<td></td>
<td>write</td>
</tr>
</tbody>
</table>
```

**Examples**

The following example shows how to set the maximum time that the LSP persists to 11,000 seconds (more than 3 hours):

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# max-lsp-lifetime 11000
```
mesh-group (IS-IS)

To optimize link-state packet (LSP) flooding in highly meshed networks, use the `mesh-group` command in interface configuration mode. To remove a subinterface from a mesh group, use the `no` form of this command.

```
mesh-group {number|blocked}
no mesh-group
```

**Syntax Description**

- `number` Number identifying the mesh group of which this interface is a member. Range is 1 to 4294967295.
- `blocked` Specifies that no LSP flooding takes place on this interface.

**Command Default**

There is no mesh group configuration (normal LSP flooding).

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

LSPs first received on subinterfaces that are not part of a mesh group are flooded to all other subinterfaces in the usual way.

LSPs first received on subinterfaces that are part of a mesh group are flooded to all interfaces except those in the same mesh group. If the `blocked` keyword is configured on a subinterface, then a newly received LSP is not flooded out over that interface.

To minimize the possibility of incomplete flooding, you should allow unrestricted flooding over at least a minimal set of links in the mesh. Selecting the smallest set of logical links that covers all physical paths results in very low flooding, but less robustness. Ideally you should select only enough links to ensure that LSP flooding is not detrimental to scaling performance, but enough links to ensure that under most failure scenarios, no router is logically disconnected from the rest of the network. In other words, blocking flooding on all links permits the best scaling performance, but there is no flooding. Permitting flooding on all links results in very poor scaling performance.

**Note**

See RFC 2973 for details about the mesh group specification.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, six interfaces are configured in three mesh groups. LSPs received are handled as follows:
• LSPs first received by GigabitEthernet interface 0/1/0/0 are flooded to all interfaces except GigabitEthernet 0/1/0/1 (which is part of the same mesh group) and GigabitEthernet 0/3/0/0 (which is blocked).

• LSPs first received by GigabitEthernet 0/2/0/0 are flooded to all interfaces except GigabitEthernet 0/2/0/0 (which is part of the same mesh group) and GigabitEthernet 0/3/0/0 (which is blocked).

• LSPs first received by GigabitEthernet 0/3/0/0 are not ignored, but flooded as usual to all interfaces.

• LSPs received first through GigabitEthernet 0/3/0/1 are flooded to all interfaces, except GigabitEthernet 0/3/0/0 (which is blocked).

RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet 0/1/0/0
RP/0/RP0/CPU0:router(config-isis-if)# mesh-group 10
RP/0/RP0/CPU0:router(config-isis-if)# exit
RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet 0/1/0/1
RP/0/RP0/CPU0:router(config-isis-if)# mesh-group 10
RP/0/RP0/CPU0:router(config-isis-if)# exit
RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet 0/2/0/0
RP/0/RP0/CPU0:router(config-isis-if)# mesh-group 11
RP/0/RP0/CPU0:router(config-isis-if)# exit
RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet 0/2/0/1
RP/0/RP0/CPU0:router(config-isis-if)# mesh-group 11
RP/0/RP0/CPU0:router(config-isis-if)# exit
RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet 0/3/0/1
RP/0/RP0/CPU0:router(config-isis-if)# mesh-group 12
RP/0/RP0/CPU0:router(config-isis-if)# exit
RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet 0/3/0/0
RP/0/RP0/CPU0:router(config-isis-if)# mesh-group blocked
metric (IS-IS)

To configure the metric for an Intermediate System-to-Intermediate System (IS-IS) interface, use the **metric** command in address family or interface address family configuration mode. To restore the default metric value, use the **no** form of this command.

```
metric {default-metric|maximum} [level {1|2}]
no metric [{default-metric|maximum} [level {1|2}]]
```

**Syntax Description**

- **default-metric**: Metric assigned to the link and used to calculate the cost from each other router using the links in the network to other destinations. Range is 1 to 63 for narrow metric and 1 to 16777214 for wide metric.

- **maximum**: Specifies maximum wide metric. All routers exclude this link from their shortest path first (SPF).

- **level**: {1 | 2} (Optional) Specifies the SPF calculation for Level 1 or Level 2 independently.

**Command Default**

**default-metric** : Default is 10.

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

Address family configuration

Interface address family configuration

**Command History**

**Release 6.0** This command was introduced.

**Usage Guidelines**

Specifying the **level** keyword resets the metric only for the specified level. We highly recommend that you configure metrics on all interfaces.

Set the default metric under address family to set the same metric for all interfaces that is associated with the address family. Set a metric value under an interface to override the default metric.

We highly recommend that you configure metrics on all interfaces.

Metrics of more than 63 cannot be used with narrow metric style.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to configure Packet-over-SONET/SDH 0/1/0/1 interface with a default link-state metric cost of 15 for Level 1:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE0/1/0/1
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# metric 15 level 1
```

The following example shows how to configure a metric cost of 15 for all interfaces under address family IPv4 unicast for level 2:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# metric 15 level 2
```
metric-style narrow

To configure the Intermediate System-to-Intermediate System (IS-IS) software to generate and accept old-style type, length, and value (TLV) objects, use the `metric-style narrow` command in address family configuration mode. To remove the `metric-style narrow` command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
metric-style narrow [transition] [level {1|2}]
no metric-style narrow [transition] [level {1|2}]
```

**Syntax Description**
- **transition** (Optional) Instructs the router to generate and accept both old-style and new-style TLV objects. It generates only old-style TLV objects.
- **level** {1 | 2} (Optional) Specifies routing Level 1 or Level 2 independently.

**Command Default**
Old-style TLVs are generated.

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**
Address family configuration

**Command History**
- Release 6.0 This command was introduced.

**Usage Guidelines**
IS-IS traffic engineering extensions include new-style TLV objects with wider metric fields than old-style TLV objects. By default, the router generates old-style TLV objects only. To perform Multiprotocol Label Switching traffic engineering (MPLS TE), a router must generate new-style TLV objects.

**Examples**
The following example shows how to configure the router to generate and accept only old-style TLV objects on router Level 1:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# metric-style narrow level 1
```
**metric-style transition**

To configure the Intermediate System-to-Intermediate System (IS-IS) software to generate and accept both old-style and new-style type, length, and value (TLV) objects, use the **metric-style transition** command in address family configuration mode. To remove the **metric-style transition** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

```
metric-style transition [level {1|2}]
no metric-style transition [level {1|2}]
```

**Syntax Description**

- **transition**
  - Instructs the router to generate and accept both old-style and new-style TLV objects.

- **level {1 | 2}**
  - (Optional) Specifies routing Level 1 or Level 2 independently.

**Command Default**

Old-style TLVs are generated, if this command is not configured.

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

Address family configuration

**Command History**

- **Release 6.0**
  - This command was introduced.

**Usage Guidelines**

IS-IS traffic engineering extensions include new-style TLV objects which have wider metric fields than old-style TLV objects. By default, the router generates old-style TLV objects only. To perform Multiprotocol Label Switching traffic engineering (MPLS TE), a router needs to generate new-style TLV objects.

**Task ID**

- **Task ID**: isis
  - **Operations**: read, write

**Examples**

The following example shows how to configure the router to generate and accept both old-style and new-style TLV objects on Level 2:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# metric-style transition level 2
```
**metric-style wide**

To configure the Intermediate System-to-Intermediate System (IS-IS) software to generate and accept only new-style type, length, and value (TLV) objects, use the `metric-style wide` command in address family configuration mode. To remove the `metric-style wide` command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
metric-style wide [transition] [level {1|2}]
no metric-style wide [transition] [level {1|2}]
```

**Syntax Description**
- `transition` (Optional) Instructs the router to generate and accept both old-style and new-style TLV objects. It generates only new-style TLV objects.
- `level {1 | 2}` (Optional) Specifies routing Level 1 or Level 2 independently.

**Command Default**
Old-style TLV lengths are generated, if this command is not configured.
Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**
Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
IS-IS traffic engineering extensions include new-style TLV objects with wider metric fields than old-style TLV objects. If you enter the `metric-style wide` command, a router generates and accepts only new-style TLV objects. Therefore, the router uses less memory and fewer other resources rather than generating both old-style and new-style TLV objects.

To perform MPLS traffic engineering, a router needs to generate new-style TLV objects.

---

**Note**
This discussion of metric styles and transition strategies is oriented toward traffic engineering deployment. Other commands and models might be appropriate if the new-style TLV objects are desired for other reasons. For example, a network may require wider metrics, but might not use traffic engineering.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to configure a router to generate and accept only new-style TLV objects on Level 1:

```
RP/0/RP0/CPU0:router(config)# router isis isp
```
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RSP0/RP0/CPU0:router(config-isis-af)# metric-style wide level 1
min-lsp-arrivaltime

To control the rate of incoming LSPs (link-state packets) LSPs, use the **min-lsp-arrivaltime** command in XR Config mode. To remove this function use the **no** form of this command.

```
min-lsp-arrivaltime [initial-wait initial] [secondary-wait secondary] [maximum-wait maximum] [level {1|2}]  
no min-lsp-arrivaltime [initial-wait initial] [secondary-wait secondary] [maximum-wait maximum] [level {1|2}]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>initial-wait</td>
<td>Initial LSP calculation delay (in milliseconds). Range is 0 to 120000.</td>
</tr>
<tr>
<td>secondary-wait</td>
<td>Hold time between the first and second LSP calculations (in milliseconds).</td>
</tr>
<tr>
<td>maximum-wait</td>
<td>Maximum interval (in milliseconds) between two consecutive LSP calculations.</td>
</tr>
<tr>
<td>level</td>
<td>(Optional) Enables the LSP interval configuration for Level 1 or Level 2 independently.</td>
</tr>
</tbody>
</table>

### Command Default

Both Level 1 and Level 2 are configured if no level is specified.

### Command Modes

XR Config mode

### Command History

**Release** | **Modification** |
---|---|
6.0 | This command was introduced.

### Usage Guidelines

This command can be used to protect a router against the possible instability of its neighbor's LSPs.

The command parameters are similar to **lsp-gen-interval** command and neighbors lsp-gen-interval values can be used to set the **min-lsp-arrivaltime**

#### Note

The initial-wait of minimum-lsp-arrival has no use in computing maximum counts and maximum window sizes of the LSP arrival time parameter.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to configure min-lsp-arrival time commands:
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config)# router isis isp min-lsp-arrivaltime
RP/0/RP0/CPU0:router(config)# router isis 1 min-lsp-arrivaltime initial-wait
RP/0/RP0/CPU0:router(config)# router isis 1 min-lsp-arrivaltime maximum-wait
RP/0/RP0/CPU0:router(config)# router isis 1 min-lsp-arrivaltime secondary-wait
mpls ldp auto-config

To enable Label Distribution Protocol (LDP) Interior Gateway Protocol (IGP) interface auto-configuration, use the `mpls ldp auto-config` command in IPv4 address family configuration mode. To disable LDP IGP auto-configuration, use the `no` form of this command.

```plaintext
mpls ldp auto-config
no mpls ldp auto-config
```

**Syntax Description**

This command has no keywords or arguments.

**Command Default**

LDP IGP auto-configuration is disabled.

**Command Modes**

IPv4 address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `mpls ldp auto-config` command to automatically configure LDP on a set of interfaces associated with a specified IGP instance. Further, LDP IGP auto-configuration provides a means to block LDP from being enabled on a specified interface. If you do not want an IS-IS interface to have LDP enabled, use the `igp auto-config disable` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to enable LDP IGP auto-configuration:

```plaintext
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# mpls ldp auto-config
```
mpls ldp sync (IS-IS)

To configure Label Distribution Protocol (LDP) IS-IS synchronization, use the `mpls ldp sync` command in interface address family configuration mode. To disable LDP synchronization, use the `no` form of this command.

```
mpls ldp sync [level {1|2}]
no mpls ldp sync [level {1|2}]
```

**Syntax Description**

| level [ {1 | 2} ] (Optional) Sets LDP synchronization for the specified level. |
|---------------------------------------------------------------|

**Command Default**

If a level is not specified, LDP synchronization is set for both levels.

**Command Modes**

Interface address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

**Note**

IS-IS advertises the maximum metric –1 (16777214) if wide metrics are configured since the maximum wide metric is specifically used for link exclusion from the shortest path first algorithm (SPF) (RFC 3784). However, the maximum narrow metric is unaffected by this definition.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read,</td>
</tr>
<tr>
<td></td>
<td>write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to enable LDP IS-IS synchronization:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# mpls ldp sync
```
nsf (IS-IS)

To enable nonstop forwarding (NSF) on the next restart, use the **nsf** command in XR Config mode. To restore the default setting, use the **no** form of this command.

```
nsf {cisco|ietf}
o nsf {cisco|ietf}
```

**Syntax Description**
- **cisco**: Specifies Cisco-proprietary NSF restart.
- **ietf**: Specifies Internet Engineering Task Force (IETF) NSF restart.

**Command Default**
NSF is disabled.

**Command Modes**
XR Config mode

**Command History**
- **Release 6.0**: This command was introduced.

**Usage Guidelines**
NSF allows an Intermediate System-to-Intermediate System (IS-IS) instance to restart using checkpointed adjacency and link-state packet (LSP) information, and to perform restart with no impact on its neighbor routers. In other words, there is no impact on other routers in the network due to the destruction and recreation of adjacencies and the system LSP.

**Task ID**
- **isis**: read, write

**Examples**
The following example shows how to enable Cisco proprietary NSF:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# nsf cisco
```
nsf interface-expire

To configure the number of resends of an acknowledged nonstop forwarding (NSF)-restart acknowledgment, use the **nsf interface-expires** command in XR Config mode. To restore the default value, use the **no** form of this command.

```
nsf interface-expires number
no nsf interface-expires
```

**Syntax Description**

- **number**: Number of resends. Range is 1 to 3.

**Command Default**

- **number**: 3 resends

**Command Modes**

- XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**

When a hello packet sent with the NSF restart flag set is not acknowledged, it is re-sent. Use the **nsf interface-expires** command to control the number of times the NSF hello is re-sent. When this limit is reached on an interface, any neighbor previously known on that interface is assumed to be down and the initial shortest path first (SPF) calculation is permitted, provided that all other necessary conditions are met.

The total time period available for adjacency reestablishment (interface-timer * interface-expires) should be greater than the expected total NSF restart time.

The **nsf interface-expires** command applies only to Internet Engineering Task Force (IETF)-style NSF. It has no effect if Cisco-proprietary NSF is configured.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to allow only one retry attempt on each interface if an IETF NSF restart signal is not acknowledged:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# nsf ietf
RP/0/RP0/CPU0:router(config-isis)# nsf interface-expires 1
```
nsf interface-timer

To configure the time interval after which an unacknowledged Internet Engineering Task Force (IETF) nonstop forwarding (NSF) restart attempt is repeated, use the **nsf interface-timer** command in router configuration mode. To restore the default value, use the **no** form of this command.

```
nsf interface-timer seconds
no nsf interface-timer
```

**Syntax Description**

- **seconds**  NSF restart time interval (in seconds). Range is 3 to 20 seconds.

**Command Default**

- **seconds**: 10 seconds

**Command Modes**

- Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When the IETF NSF restart process begins, hello packets send an NSF restart flag that must be acknowledged by the neighbors of the router. Use the **nsf interface-timer** command to control the restart time interval after the hello packet is re-sent. The restart time interval need not match the hello interval.

The **nsf interface-timer** command applies only to IETF-style NSF. It has no effect if Cisco proprietary NSF is configured.

**Examples**

The following example shows how to ensure that a hello packet with the NSF restart flag set is sent again every 5 seconds until the flag is acknowledged:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# nsf ietf
RP/0/RP0/CPU0:router(config-isis)# nsf interface-timer 5
```
**nsf lifetime (IS-IS)**

To configure the maximum route lifetime following a nonstop forwarding (NSF) restart, use the `nsf lifetime` command in XR Config mode. To restore the default value, use the `no` form of this command.

```
nsf lifetime seconds
no nsf lifetime
```

**Syntax Description**

- `seconds` Maximum route lifetime (in seconds) following an NSF restart. Range is 5 to 300 seconds.

**Command Default**

- `seconds`: 60 seconds (1 minute)

**Command Modes**

- XR Config mode

**Command History**

- **Release** Modification
  - Release 6.0 This command was introduced.

**Usage Guidelines**

Use the `nsf lifetime` command to set the maximum available time for the reacquisition of checkpointed adjacencies and link-state packets (LSPs) during a Cisco proprietary NSF restart. LSPs and adjacencies not recovered during this time period are abandoned, thus causing changes to the network topology.

**Examples**

The following example shows how to configure the router to allow only 20 seconds for the entire NSF process:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# nsf cisco
RP/0/RP0/CPU0:router(config-isis)# nsf lifetime 20
```
passive (IS-IS)

To suppress Intermediate System-to-Intermediate System (IS-IS) packets from being transmitted to the interface and received packets from being processed on the interface, use the `passive` command in interface configuration mode. To restore IS-IS packets coming to an interface, use the `no` form of this command.

```
passive
no passive
```

**Command Default**
Interface is active.

**Command Modes**
Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to configure the router to suppress IS-IS packets on GigabitEthernet interface 0/1/0/1:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface GigabitEthernet 0/1/0/1
RP/0/RP0/CPU0:router(config-isis-if)# passive
```
point-to-point

To configure a network of only two networking devices that use broadcast media and the integrated Intermediate System-to-Intermediate System (IS-IS) routing protocol to function as a point-to-point link instead of a broadcast link, use the point-to-point command in interface configuration mode. To disable the point-to-point usage, use the no form of this command.

point-to-point
no point-to-point

Syntax Description
This command has no keywords or arguments.

Command Default
Interface is treated as broadcast if connected to broadcast media.

Command Modes
Interface configuration

Command History
Release Modification
Release 6.0 This command was introduced.

Usage Guidelines
Use the point-to-point command only on broadcast media in a network with two networking devices. The command causes the system to issue packets point-to-point rather than as broadcasts. Configure the command on both networking devices in the network.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples
The following example shows how to configure a 10-Gb Ethernet interface to act as a point-to-point interface:

```
RP/0/RP0/CPU0:router(config)#  router isis isp
RP/0/RP0/CPU0:router(config-isis)#  interface TenGigE 0/6/0/0
RP/0/RP0/CPU0:router(config-isis-if)#  point-to-point
```
prefix-sid index

To specify or advertize prefix (node) segment ID (SID) on all routers, use the `prefix-sid index` command in IPv4 address family configuration mode. To stop advertizing prefix SID, use the `no` form of this command.

The segment routing must be configured on the ISIS instance before configuring prefix SID value.

```
prefix-sid index  sid-value

no prefix-sid index  sid-value
```

**Syntax Description**

- `sid-value` Specifies the prefix SID value. Value range is between 0 and 1048575.

**Command Default**

No default behavior or values.

**Command Modes**

IPv4 address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task</th>
<th>Operations</th>
<th>ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
<td></td>
</tr>
</tbody>
</table>

**Examples**

This example shows how to advertize prefix SID.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router isis 100
RP/0/RP0/CPU0:router(config-isis)# interface loopback0
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# prefix-sid index 16041
```
priority (IS-IS)

To configure the priority of designated routers, use the `priority` command in interface configuration mode. To reset the default priority, use the `no` form of this command.

```
priority value [level {1|2}]
no priority [value] [level {1|2}]
```

**Syntax Description**

- `value` (Optional) Specifies routing Level 1 or Level 2 independently.
- `level { 1 | 2 }` (Optional) Specifies routing Level 1 or Level 2 independently.

**Command Default**

- `value : 64`
- `Both Level 1 and Level 2 are configured if no level is specified.`

**Command Modes**

- `Interface configuration`

**Command History**

- `Release 6.0` This command was introduced.

**Usage Guidelines**

Priorities can be configured for Level 1 and Level 2 independently. Specifying Level 1 or Level 2 resets priority only for Level 1 or Level 2 routing, respectively. Specifying no level allows you to configure all levels.

The priority is used to determine which router on a LAN is the designated router or Designated Intermediate System (DIS). The priorities are advertised in the hello packets. The router with the highest priority becomes the DIS.

In the Intermediate System-to-Intermediate System (IS-IS) protocol, there is no backup designated router. Setting the priority to 0 lowers the chance of this system becoming the DIS, but does not prevent it. If a router with a higher priority comes online, it takes over the role from the current DIS. For equal priorities, the higher MAC address breaks the tie.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to give Level 1 routing priority by setting the priority level to 80. This router is now more likely to become the DIS.

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface TenGigE 0/6/0/0
RP/0/RP0/CPU0:router(config-isis-if)# priority 80 level 1
```
propagate level

To propagate routes from one Intermediate System-to-Intermediate System (IS-IS) level into another level, use the `propagate level` command in address family configuration mode. To disable propagation, use the `no` form of this command.

```
propagate level {1|2} into level {1|2} route-policy route-policy-name
no propagate level {1|2} into level {1|2}
```

**Syntax Description**

- `level {1 | 2}`: Propagates from routing Level 1 or Level 2 routes.
- `into`: Propagates from Level 1 or Level 2 routes into Level 1 or Level 2 routes.
- `route-policy route-policy-name`: Specifies a configured route policy.

**Command Default**

Route leaking (Level 2 to Level 1) is disabled.

**Command Modes**

Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

In general, route propagation from Level 1 to Level 2 is automatic. You might want to use this command to better control which Level 1 routes can be propagated into Level 2.

Propagating Level 2 routes into Level 1 is called `route leaking`. Route leaking is disabled by default. That is, Level 2 routes are not automatically included in Level 1 link-state packets (LSPs). If you want to leak Level 2 routes into Level 1, you must enable that behavior by using this command.

Propagation from Level 1 into Level 1 and from Level 2 into Level 2 is not allowed.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to redistribute Level 2 routes to Level 1:

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list 101 permit ip 10.0.0.0 255.0.0.0 10.1.0.1 0.255.255.255
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# net 49.1234.2222.2222.2222.00
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# propagate level 2 into level 1 route-policy policy_a
```
redistribute (IS-IS)

To redistribute routes from one routing protocol into Intermediate System-to-Intermediate System (IS-IS), use the redistribute command in address family configuration mode. To remove the redistribute command from the configuration file and restore the system to its default condition in which the software does not redistribute routes, use the no form of this command.

Border Gateway Protocol (BGP)
redistribute bgp process-id [{level-1|level-2|level-1-2}] [metric metric-value] [metric-type {internal|external|rib-metric-as-external|rib-metric-as-internal}] [route-policy route-policy-name]
no redistribute

Connected Routes
redistribute connected [{level-1|level-2|level-1-2}] [metric metric-value] [metric-type {internal|external|rib-metric-as-external|rib-metric-as-internal}] [route-policy route-policy-name]
no redistribute

Intermediate System-to-Intermediate System (IS-IS)
redistribute isis process-id [{level-1|level-2|level-1-2}] [metric metric-value] [metric-type {internal|external|rib-metric-as-external|rib-metric-as-internal}] [route-policy route-policy-name]
no redistribute

Open Shortest Path First (OSPF)
redistribute ospf process-id [{level-1|level-2|level-1-2}] [match {external [{1|2}]|internal|nssa-external [{1|2}]}] [metric metric-value] [metric-type {internal|external|rib-metric-as-external|rib-metric-as-internal}] [route-policy route-policy-name]
no redistribute

Open Shortest Path First Version 3 (OSPFv3)
redistribute ospfv3 process-id [{level-1|level-2|level-1-2}] [match {external [{1|2}]|internal|nssa-external [{1|2}]}] [metric metric-value] [metric-type {internal|external|rib-metric-as-external|rib-metric-as-internal}] [route-policy route-policy-name]
no redistribute

Static Routes
redistribute static [{level-1|level-2|level-1-2}] [metric metric-value] [metric-type {1 {internal|external|rib-metric-as-external} 2 rib-metric-as-internal}] [route-policy route-policy-name]
no redistribute
<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>process-id</strong></td>
<td>For the <strong>bgp</strong> keyword, an autonomous system number has the following ranges:</td>
</tr>
<tr>
<td></td>
<td>• Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.</td>
</tr>
<tr>
<td></td>
<td>• Range for 4-byte Autonomous system numbers (ASNs) in a plain format is 1 to 4294967295.</td>
</tr>
<tr>
<td></td>
<td>• Range for 4-byte Autonomous system numbers (ASNs) in a dot format is 1.0 to 65535.65535.</td>
</tr>
<tr>
<td></td>
<td>For the <strong>isis</strong> keyword, an IS-IS instance identifier from which routes are to be redistributed.</td>
</tr>
<tr>
<td></td>
<td>For the <strong>ospf</strong> keyword, an OSPF process name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.</td>
</tr>
<tr>
<td></td>
<td>For the <strong>ospfv3</strong> keyword, an OSPFv3 process name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.</td>
</tr>
<tr>
<td><strong>level-1</strong></td>
<td>(Optional) Specifies that redistributed routes are advertised in the Level-1 LSP of the router.</td>
</tr>
<tr>
<td><strong>level-1-2</strong></td>
<td>(Optional) Specifies that redistributed routes are advertised in the Level-1-2 LSP of the router.</td>
</tr>
<tr>
<td><strong>level-2</strong></td>
<td>(Optional) Specifies that redistributed routes are advertised in the Level-2 LSP of the router.</td>
</tr>
<tr>
<td><strong>metric metric-value</strong></td>
<td>(Optional) Specifies the metric used for the redistributed route. Range is 0 to 16777215. The <strong>metric-value</strong> must be consistent with the IS-IS metric style of the area and topology into which the routes are being redistributed.</td>
</tr>
<tr>
<td><strong>metric-type</strong></td>
<td>(Optional) Specifies the external link type associated with the route advertised into the ISIS routing domain. It can be one of two four values:</td>
</tr>
<tr>
<td></td>
<td>• external</td>
</tr>
<tr>
<td></td>
<td>• <strong>internal</strong> – Use the <strong>internal</strong> keyword to set IS-IS internal metric-type</td>
</tr>
<tr>
<td></td>
<td>• <strong>external</strong> – Use the <strong>external</strong> keyword to set IS-IS external metric-type</td>
</tr>
<tr>
<td></td>
<td>• <strong>rib-metric-as-external</strong> – Use the <strong>rib-metric-as-external</strong> keyword to use RIB metric and set IS-IS external metric-type</td>
</tr>
<tr>
<td></td>
<td>• <strong>rib-metric-as-internal</strong> – Use the <strong>rib-metric-as-internal</strong> keyword to use RIB metric and set IS-IS internal metric-type</td>
</tr>
<tr>
<td></td>
<td>Any route with an internal metric (however large the metric is) is preferred over a route with external metric (however small the metric is).</td>
</tr>
<tr>
<td></td>
<td>Use the <strong>rib-metric-as-external</strong> and <strong>rib-metric-as-internal</strong> keywords to preserve RIB metrics when redistributing routes from another IS-IS router instance or another protocol.</td>
</tr>
<tr>
<td><strong>route-policy</strong></td>
<td>(Optional) Specifies the identifier of a configured policy. A policy is used to filter the importation of routes from this source routing protocol to IS-IS.</td>
</tr>
<tr>
<td><strong>route-policy-name</strong></td>
<td></td>
</tr>
</tbody>
</table>

Routing Command Reference for Cisco NCS 5000 Series Routers
match { internal | external [ 1 | 2 ] | nsaa-external [ 1 | 2 ]} (Optional) Specifies the criteria by which OSPF routes are redistributed into other routing domains. It can be one or more of the following:

- **internal** — Routes that are internal to a specific autonomous system (intra- and interarea OSPF routes).
- **external [ 1 | 2 ]** — Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 external routes.
- **nsaa-external [ 1 | 2 ]** — Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 not-so-stubby area (NSSA) external routes.

For the external and nsaa-external options, if a type is not specified, then both Type 1 and Type 2 are assumed.

**Command Default**

Level 2 is configured if no level is specified.

**metric-type: internal**

**match** : If no match keyword is specified, all OSPF routes are redistributed.

**Command Modes**

Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

**Note**

When redistributing routes (into IS-IS) using both command keywords for setting or matching of attributes and a route policy, the routes are run through the route policy first, followed by the keyword matching and setting.

Use the redistribute command to control the redistribution of routes between separate IS-IS instances. To control the propagation of routes between the levels of a single IS-IS instance, use the propagate level, on page 68 command.

Only IPv4 OSPF addresses can be redistributed into IS-IS IPv4 address families and only IPv6 OSPFv3 prefixes can be distributed into IS-IS IPv6 address families.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In this example, IS-IS instance isp_A readvertises all of the routes of IS-IS instance isp_B in Level 2 LSP. Note that the level-2 keyword affects which levels instance isp_A advertises the routes in and has no impact on which routes from instance isp_B are advertised. (Any Level 1 routes from IS-IS instance isp_B are included in the redistribution.)
RP/0/RSP00/CPU0:router(config)# router isis isp_A
RP/0/RSP00/CPU0:router(config-isis)# net 49.1234.2222.2222.2222.00
RP/0/RSP00/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RSP00/CPU0:router(config-isis-af)# redistribute isis isp_B level-2
!
RP/0/RSP00/CPU0:router(config)# router isis isp_B
RP/0/RSP00/CPU0:router(config-isis)# is-type level 1
RP/0/RSP00/CPU0:router(config-isis)# net 49.4567.2222.2222.2222.00
RP/0/RSP00/CPU0:router(config-isis)# address-family ipv4 unicast
retransmit-interval (IS-IS)

To configure the amount of time between retransmission of each Intermediate System-to-Intermediate System (IS-IS) link-state packet (LSP) on a point-to-point link, use the `retransmit-interval` command in interface configuration mode. To restore the default value, use the `no` form of this command.

```
retransmit-interval seconds [level {1|2}]
no retransmit-interval [seconds [level {1|2}]]
```

**Syntax Description**

- `seconds` Time (in seconds) between consecutive retransmissions of each LSP. It is an integer that should be greater than the expected round-trip delay between any two networking devices on the attached network. Range is 0 to 65535 seconds.

- `level {1 | 2}` (Optional) Specifies routing Level 1 or Level 2 independently.

**Command Default**

`seconds : 5 seconds`

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Release 6.0</td>
</tr>
<tr>
<td></td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

To prevent needless transmission results, the `seconds` argument should be conservative.

The `retransmit-interval` command has no effect on LAN (multipoint) interfaces. On point-to-point links, the value can be increased to enhance network stability.

Because retransmissions occur only when LSPs are dropped, setting this command to a higher value has little effect on reconvergence. The more neighbors networking devices have, and the more paths over which LSPs can be flooded, the higher this value can be made.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure GigabitEthernet interface 0/2/0/1 for retransmission of IS-IS LSPs every 60 seconds for a large serial line:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-isis-if)# retransmit-interval 60
```
retransmit-throttle-interval

To configure minimum interval between retransmissions of different Intermediate System-to-Intermediate System (IS-IS) link-state packets (LSPs) on a point-to-point interface, use the `retransmit-throttle-interval` command in interface configuration mode. To remove the command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
retransmit-throttle-interval  milliseconds  [level  {1|2}]
no  retransmit-throttle-interval  [milliseconds  [level  {1|2}]]
```

### Syntax Description

- **milliseconds**: Minimum delay (in milliseconds) between LSP retransmissions on the interface. Range is 0 to 65535.
- **level**: `{1 | 2}` (Optional) Specifies routing Level 1 or Level 2 independently.

### Command Default

Default is 0.

### Command Modes

Interface configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `retransmit-throttle-interval` command to define the minimum period of time that must elapse between retransmitting any two consecutive LSPs on an interface. The `retransmit-throttle-interval` command may be useful in very large networks with many LSPs and many interfaces as a way of controlling LSP retransmission traffic. This command controls the rate at which LSPs can be re-sent on the interface.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to configure tenGigE interface 0/2/0/1 to limit the rate of LSP retransmissions to one every 300 milliseconds:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-isis-if)# retransmit-throttle-interval 300
```
route source first-hop

To replace the originating route with first-hop for multicast traffic, use the route source first-hop command in ISIS address-family submode. To remove the first-hop for multicast traffic, use the no form of this command.

route source first-hop

This command has no keywords or arguments.

Command Default

no route source first-hop is enabled.

Command Modes

ISIS address-family submode

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

This command replaces the originating router address with first-hop router address in the RIB table and facilitates computing alternate paths for multicast traffic. This feature is incompatible with other IOS-XR features, such as MPLS-TE inter-area tunnels. You must use the route source first-hop command only to support MoFRR with multicast multipath.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples

The following example shows how to replace the originating route with first-hop:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 multicast
RP/0/RP0/CPU0:router(config-isis-af)# route source first-hop
```

Related Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>net</td>
<td>Configures an IS-IS NET for the routing process.</td>
</tr>
</tbody>
</table>
**segment-routing**

To enable segment routing for IPv4 addresses with MPLS data plane, use the `segment-routing` command in IPv4 address family configuration mode. To disable segment routing, use the `no` form of this command.

```
segment-routing mpls

no segment-routing
```

**Syntax Description**

- `mpls`: Enables segment routing for IPv4 addresses with MPLS data plane.

**Command Default**

No default behavior or values

**Command Modes**

IPv4 address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The prefix SID value must be removed from all the interfaces under the same ISIS instance before disabling segment routing.

**Examples**

This example shows how to enable segment routing with MPLS data plane.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router isis 100
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# segment-routing mpls
```
set-attached-bit

To configure an Intermediate System-to-Intermediate System (IS-IS) instance with an attached bit in the Level 1 link-state packet (LSP), use the `set-attached-bit` command in address family configuration mode. To remove the `set-attached-bit` command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
set-attached-bit
no set-attached-bit
```

**Command Default**
Attached bit is not set in the LSP.

**Command Modes**
Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set-attached-bit` command to set an IS-IS instance with an attached bit in the Level 1 LSP that allows another IS-IS instance to redistribute Level 2 topology. The attached bit is used when the Level 2 connectivity from another IS-IS instance is advertised by the Level 1 attached bit.

Cisco IOS XR software does not support multiple Level 1 areas in a single IS-IS routing instance. But the equivalent functionality is achieved by redistribution of routes between two IS-IS instances by using the `redistribute (IS-IS)`, on page 69 command.

The attached bit is configured for a specific address family only if the `single-topology` command is not configured.

**Note**
If connectivity for the Level 2 instance is lost, the attached bit in the Level 1 instance LSP continues sending traffic to the Level 2 instance and causes the traffic to be dropped.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the attached bit for a Level 1 instance that allows the Level 2 instance to redistribute routes from the Level 1 instance:

```
RP/0/RP0/CPU0:router(config)# router isis 1
RP/0/RP0/CPU0:router(config-isis)# net 49.0001.0001.0001.0001.0001.0001.00
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# redistribute isis 2 level 2
! RP/0/RP0/CPU0:router(config-isis-af)# interface tenGigE 0/3/0/0
```
RP/0/RP0/CPU0:router(config-isis-af-if)# address-family ipv4 unicast
!
RP/0/RP0/CPU0:router(config)# router isis 2
RP/0/RP0/CPU0:router(config)# is-type level-1
RP/0/RP0/CPU0:router(config)# net 49.0002.0001.0001.0002.00
RP/0/RP0/CPU0:router(config)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# set-attached-bit send always-bitset
!
RP/0/RP0/CPU0:router(config-isis-af)# interface tenGigE 0/1/0/0
RP/0/RP0/CPU0:router(config-isis-af-if)# address-family ipv4 unicast
**set-overload-bit**

To configure the router to signal other routers not to use it as an intermediate hop in their shortest path first (SPF) calculations, use the `set-overload-bit` command in XR Config mode. To remove the designation, use the `no` form of this command.

```
set-overload-bit [on-startup {delay|wait-for-bgp}] [level {1|2}] [advertise {external|interlevel}]
no set-overload-bit [on-startup {delay|wait-for-bgp}] [level {1|2}] [advertise {external|interlevel}]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>on-startup</code></td>
<td>(Optional) Sets the overload bit only temporarily after reboot.</td>
</tr>
<tr>
<td><code>delay</code></td>
<td>(Optional) Time (in seconds) to advertise when the router is overloaded after reboot. Range is 5 to 86400 seconds (86400 seconds = 1 day).</td>
</tr>
<tr>
<td><code>wait-for-bgp</code></td>
<td>(Optional) Sets the overload bit on startup until the Border Gateway Protocol (BGP) signals converge or time out.</td>
</tr>
<tr>
<td>`level {1</td>
<td>2}`</td>
</tr>
<tr>
<td>`advertise {external</td>
<td>interlevel}`</td>
</tr>
<tr>
<td></td>
<td>• external—If overload-bit set advertises IP prefixes learned from other protocols</td>
</tr>
<tr>
<td></td>
<td>• interlevel—If overload-bit set advertise IP prefixes learned from another IS-IS level</td>
</tr>
</tbody>
</table>

### Command Default

The overload bit is not set.

Both Level 1 and Level 2 are configured if no level is specified.

### Command Modes

XR Config mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `set-overload-bit` command to force the router to set the overload bit in its nonpseudonode link-state packets (LSPs). Normally the setting of the overload bit is allowed only when a router experiences problems. For example, when a router is experiencing a memory shortage, the reason might be that the link-state database is not complete, resulting in an incomplete or inaccurate routing table. If the overload bit is set in the LSPs of the unreliable router, other routers can ignore the router in their SPF calculations until it has recovered from its problems. The result is that no paths through the unreliable router are seen by other routers in the Intermediate System-to-Intermediate System (IS-IS) area. However, IP prefixes directly connected to this router are still reachable.

The `set-overload-bit` command can be useful when you want to connect a router to an IS-IS network, but do not want real traffic flowing through it under any circumstances.

Routers with overload bit set are:
• A test router in the lab, connected to a production network.
• A router configured as an LSP flooding server, for example, on a nonbroadcast multiaccess (NBMA) network, in combination with the mesh group feature.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

Examples

The following example shows how to configure the overload bit:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# set-overload-bit
```
show isis

The show isis command displays general information about an IS-IS instance and protocol operation. If the instance ID is not specified, the command shows information about all IS-IS instances.

`show isis [instance instance-id]`

**Syntax Description**

- `instance instance-id` (Optional) Displays the IS-IS adjacencies for the specified IS-IS instance only.

  **Note** The instance-id argument is the instance identifier (alphanumeric) defined by the `router isis` command.

**Command Default**

No instance ID specified displays IS-IS adjacencies for all the IS-IS instances.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

For each instance, the first line of output lists the IS-IS instance ID with the following lines identifying the IS-IS system ID, supported levels (level 1, level 2, or level-1-2), configured area addresses, active area addresses, status (enabled or not) and type (Cisco or IETF) of nonstop forwarding (NSF), and the mode in which the last IS-IS process startup occurred.

Next, the status of each configured address family (or just IPv4 unicast if none are configured) is summarized. For each level (level 1 or level 2), the metric style (narrow or wide) generated and accepted is listed along with the status of incremental shortest path first (iSPF) computation (enabled or not). Then redistributed protocols are listed, followed by the administrative distance applied to the redistributed routes.

Finally, the running state (active, passive, or disabled) and configuration state (active or disabled) of each IS-IS interface is listed.

**Examples**

The following is sample output from the `show isis` command:

```
RP/0/RP0/CPU0:router# show isis
Wed Aug 20 23:54:55.043 PST DST
IS-IS Router: lab
    System Id: 0000.0000.0002
    IS Levels: level-2-only
    Manual area address(es):
        49.1122
    Routing for area address(es):
        49.1122
```
Non-stop forwarding: Disabled  
Most recent startup mode: Cold Restart  
Topologies supported by IS-IS:  
   IPv4 Unicast  
      Level-2  
         Metric style (generate/accept): Narrow/Narrow  
         Metric: 10  
         ISPF status: Disabled  
         No protocols redistributed  
         Distance: 115  
Interfaces supported by IS-IS:  
   Loopback0 is running passively (passive in configuration)  
   POS0/1/0/2 is running actively (active in configuration)  
   POS0/1/0/3 is running actively (active in configuration)

This table describes the significant fields shown in the display.

**Table 3: show isis Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IS-IS Router</td>
<td>IS-IS instance ID.</td>
</tr>
<tr>
<td>System Id</td>
<td>IS-IS system ID.</td>
</tr>
<tr>
<td>IS Levels</td>
<td>Supported system levels for the instance.</td>
</tr>
<tr>
<td>Manual area address(es)</td>
<td>Domain and area.</td>
</tr>
<tr>
<td>Routing for area address(es):</td>
<td>Configured area addresses and active area addresses.</td>
</tr>
<tr>
<td>Non-stop forwarding</td>
<td>Status (enabled or not) and type (Cisco or IETF) of nonstop forwarding (NSF).</td>
</tr>
<tr>
<td>Most recent startup mode</td>
<td>The mode in which the last IS-IS process startup occurred.</td>
</tr>
<tr>
<td>Topologies supported by IS-IS</td>
<td>The summary of the status of each configured address family (or just IPv4 unicast if none are configured).</td>
</tr>
<tr>
<td>Distributed protocols</td>
<td>List of redistributed protocols, followed by the administrative distance applied to the redistributed routes.</td>
</tr>
<tr>
<td>Metric style (generate/accept)</td>
<td>The status of each configured address family (or just IPv4 unicast if none are configured) is summarized. For each level (level 1 or level 2), the metric style (narrow or wide) generated and accepted is listed along with the status of incremental shortest path first (iSPF) computation (enabled or not).</td>
</tr>
<tr>
<td>Interfaces supported by IS-IS</td>
<td>The running state (active, passive, or disabled) and configuration state (active or disabled) of each IS-IS interface.</td>
</tr>
</tbody>
</table>
show isis adjacency

To display Intermediate System-to-Intermediate System (IS-IS) adjacencies, use the `show isis adjacency` command in XR EXEC mode.

```
show isis [instance instance-id] adjacency [level {1|2}] [type interface-path-id] [detail] [systemid system-id]
```

**Syntax Description**

- **instance instance-id** (Optional) Displays the IS-IS adjacencies for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

- **level {1 2]** (Optional) Displays the IS-IS adjacencies for Level 1 or Level 2 independently.

- **type** Interface type. For more information, use the question mark ( ? ) online help function.

- **interface-path-id** Physical interface or virtual interface.

  **Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

- **detail** (Optional) Displays neighbor IP addresses and active topologies.

- **systemid system-id** (Optional) Displays the information for the specified router only.

**Command Default**

No instance ID specified displays IS-IS adjacencies for all the IS-IS instances.

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

XR EXEC mode

**Command History**

- **Release Modification**
  - Release 6.0 This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- **Task Operations ID**
  - isis read

**Examples**

The following is sample output from the `show isis adjacency` command:

```
RP/0/RP0/CPU0:router# show isis adjacency
```
IS-IS p Level-1 adjacencies:

<table>
<thead>
<tr>
<th>System Id</th>
<th>Interface</th>
<th>SNPA</th>
<th>State</th>
<th>Hold</th>
<th>Changed</th>
<th>NSF</th>
<th>BFD</th>
</tr>
</thead>
<tbody>
<tr>
<td>12a4</td>
<td>PO0/1/0/1</td>
<td><em>PtoP</em></td>
<td>Up</td>
<td>23</td>
<td>00:00:06</td>
<td>Capable</td>
<td>Init</td>
</tr>
<tr>
<td>12a4</td>
<td>Gi0/6/0/2</td>
<td>0004.2893.f2f6</td>
<td>Up</td>
<td>56</td>
<td>00:04:01</td>
<td>Capable</td>
<td>Up</td>
</tr>
</tbody>
</table>

Total adjacency count: 2

IS-IS p Level-2 adjacencies:

<table>
<thead>
<tr>
<th>System Id</th>
<th>Interface</th>
<th>SNPA</th>
<th>State</th>
<th>Hold</th>
<th>Changed</th>
<th>NSF</th>
<th>BFD</th>
</tr>
</thead>
<tbody>
<tr>
<td>12a4</td>
<td>PO0/1/0/1</td>
<td><em>PtoP</em></td>
<td>Up</td>
<td>23</td>
<td>00:00:06</td>
<td>None</td>
<td></td>
</tr>
<tr>
<td>12a4</td>
<td>Gi0/6/0/2</td>
<td>0004.2893.f2f6</td>
<td>Up</td>
<td>26</td>
<td>00:00:13</td>
<td>Capable</td>
<td>Init</td>
</tr>
</tbody>
</table>

Total adjacency count: 2

This table describes the significant fields shown in the display.

**Table 4: show isis adjacency Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level-1</td>
<td>Level 1 adjacencies.</td>
</tr>
<tr>
<td>Level-2</td>
<td>Level 2 adjacencies.</td>
</tr>
<tr>
<td>System ID</td>
<td>Dynamic hostname of the system. The hostname is specified using the <code>hostname</code> command. If the dynamic hostname is not known or the <code>hostname dynamic disable</code> command has been executed, the 6-octet system ID is used.</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface used to reach the neighbor.</td>
</tr>
<tr>
<td>SNPA</td>
<td>Data-link address (also known as the Subnetwork Point of Attachment [SNPA]) of the neighbor.</td>
</tr>
<tr>
<td>State</td>
<td>Adjacency state of the neighboring interface. Valid states are Down, Init, and Up.</td>
</tr>
<tr>
<td>Holdtime</td>
<td>Hold time of the neighbor.</td>
</tr>
<tr>
<td>Changed</td>
<td>Time the neighbor has been up (in hours:minutes:seconds).</td>
</tr>
<tr>
<td>NSF</td>
<td>Specifies whether the neighbor can adhere to the IETF-NSF restart mechanism.</td>
</tr>
<tr>
<td>BFD</td>
<td>Specifies the Bidirectional Forwarding Detection (BFD) status for the interface. Valid status are:</td>
</tr>
<tr>
<td></td>
<td>• None—BFD is not configured.</td>
</tr>
<tr>
<td></td>
<td>• Init—BFD session is not up. One reason is that other side is not yet enabled.</td>
</tr>
<tr>
<td></td>
<td>• Up—BFD session has been established.</td>
</tr>
<tr>
<td></td>
<td>• Down—BFD session holdtime expired.</td>
</tr>
</tbody>
</table>
**show isis adjacency-log**

To display the Intermediate System-to-Intermediate System (IS-IS) adjacency log, use the **show isis adjacency-log** command in XR EXEC mode.

```
show isis adjacency-log [level {1|2}] [{last number|first number}]
```

**Syntax Description**

- `level {1|2}` (Optional) Displays the IS-IS adjacency log for Level 1 or Level 2 independently.
- `last number` (Optional) Specifies that the output is restricted to the last `number` of entries. Range is 1 to 100.
- `first number` (Optional) Specifies that the output is restricted to the first `number` of entries. Range is 1 to 100.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- **Task**
  - read

**Examples**

The following is sample output from the **show isis adjacency-log** command:

```
RP/0/RP0/CPU0:router# show isis adjacency-log

IS-IS 10 Level 1 Adjacency log
When  System  Interface  State               Details
4d00h  12a1  PO0/5/0/0  d -> i
4d00h  12a1  PO0/5/0/0  i -> u  New adjacency
        4d00h  12a1  PO0/5/0/0  IPv4 Unicast Up
4d00h  12a1  Gi0/6/0/0  d -> u  New adjacency
4d00h  12a1  Gi0/6/0/0  u -> d  Interface state
down
3d17h  12a1  Gi0/6/0/0  d -> u  New adjacency
down
3d17h  12a1  Gi0/6/0/0  u -> d  Interface state
down
01:44:07  12a1  Gi0/6/0/0  d -> u  New adjacency

IS-IS 10 Level 2 Adjacency log
When  System  Interface  State               Details
4d00h  12a1  PO0/5/0/0  d -> i
4d00h  12a1  PO0/5/0/0  i -> u  New adjacency
```
IPv4 Unicast Up

<table>
<thead>
<tr>
<th>Time</th>
<th>SystemID</th>
<th>Interface</th>
<th>State Transition</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4d00h</td>
<td>12a1</td>
<td>G1/0/6/0/0</td>
<td>d -&gt; u</td>
<td>New adjacency</td>
</tr>
<tr>
<td>down</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3d17h</td>
<td>12a1</td>
<td>G1/0/6/0/0</td>
<td>u -&gt; d</td>
<td>Interface state</td>
</tr>
<tr>
<td>down</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01:44:07</td>
<td>12a1</td>
<td>G1/0/6/0/0</td>
<td>d -&gt; u</td>
<td>New adjacency</td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

**Table 5: show isis adjacency-log Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>When</td>
<td>Elapsed time (in hh:mm:ss) since the event was logged.</td>
</tr>
<tr>
<td>System</td>
<td>System ID of the adjacent router.</td>
</tr>
<tr>
<td>Interface</td>
<td>Specific interface involved in the adjacency change.</td>
</tr>
<tr>
<td>State</td>
<td>State transition for the logged event.</td>
</tr>
<tr>
<td>Details</td>
<td>Description of the adjacency change.</td>
</tr>
</tbody>
</table>
show isis checkpoint adjacency

To display the Intermediate System-to-Intermediate System (IS-IS) checkpoint adjacency database, use the `show isis checkpoint adjacency` command in XR EXEC mode.

```
show isis [instance instance-id] checkpoint adjacency
```

**Syntax Description**

- `instance instance-id` (Optional) Displays the IS-IS checkpoint adjacencies for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

**Command Default**

No instance ID specified displays IS-IS checkpoint adjacencies for all the IS-IS instances.

**Command Modes**

XR EXEC mode

**Command History**

Release 6.0 This command was introduced.

**Usage Guidelines**

Use the `show isis checkpoint adjacency` command to display the checkpointed adjacencies. With this information you can restore the adjacency database during a Cisco proprietary nonstop forwarding (NSF) restart. This command, with the `show isis adjacency` command, can be used to verify the consistency of the two databases.

**Task ID**

- **Task ID**: isis
- **Operations ID**: read

**Examples**

The following is sample output from the `show isis checkpoint adjacency` command:

```
RP/0/RP0/CPU0:router# show isis checkpoint adjacency

Interface   Level  System ID            State Circuit ID           Chkpt ID
Gi3/0/0/1   1      router-gsr8        Up    0001.0000.0008.04    80011fec
Gi0/4/0/1   1      router-gsr9        Up    0001.0000.0006.01    80011fd8
Gi3/0/0/1   2      router-gsr8        Up    0001.0000.0008.04    80011fc4
```
This table describes the significant fields shown in the display.

**Table 6: show isis checkpoint adjacency Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>Interface used to reach the neighbor.</td>
</tr>
<tr>
<td>Level</td>
<td>Lists either routers with Level 1 or Level 2 adjacency configured.</td>
</tr>
<tr>
<td>System ID</td>
<td>Dynamic hostname of the system. The hostname is specified using the <code>hostname</code> command. If the dynamic hostname is not known or <code>hostname dynamic disable</code> command has been executed, the 6-octet system ID is used.</td>
</tr>
<tr>
<td>State</td>
<td>State of the neighboring interface.</td>
</tr>
<tr>
<td>Circuit ID</td>
<td>Unique ID issued to a circuit at its creation.</td>
</tr>
<tr>
<td>Chkpt ID</td>
<td>Unique ID issued to the checkpoint at its creation.</td>
</tr>
</tbody>
</table>
show isis checkpoint interface

To display the Intermediate System-to-Intermediate System (IS-IS) checkpoint interfaces, use the show isis checkpoint interface command in XR EXEC mode.

show isis checkpoint interface

This command has no keywords or arguments.

Command Default
No default behavior or values

Command Modes
XR EXEC mode

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines
No specific guidelines impact the use of this command.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the show isis checkpoint interface command:

```
RP/0/RP0/CPU0:router# show isis checkpoint interface

IS-IS 10 checkpoint interface
Interface    Index    CircNum    DIS Areas    Chkpt ID
Po0/5/0/0    0         0          NONE         80002fe8
Gi0/6/0/0    1         3          L1L2         80002fd0
```

This table describes the significant fields shown in the display.

Table 7: show isis checkpoint interface Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>Interface used to reach the neighbor.</td>
</tr>
<tr>
<td>Index</td>
<td>Interface index assigned to an interface upon its creation.</td>
</tr>
<tr>
<td>CircNum</td>
<td>Unique ID issued to a circuit internally.</td>
</tr>
<tr>
<td>DIS Areas</td>
<td>Designated Intermediate System area.</td>
</tr>
<tr>
<td>Chkpt ID</td>
<td>Unique ID issued to the checkpoint at its creation.</td>
</tr>
</tbody>
</table>
show isis checkpoint lsp

To display the Intermediate System-to-Intermediate System (IS-IS) checkpoint link-state packet (LSP) protocol data unit (PDU) identifier database, use the `show isis checkpoint lsp` command in XR EXEC mode.

```
show isis [instance instance-id] checkpoint lsp
```

**Syntax Description**

- `instance instance-id` (Optional) Displays the IS-IS checkpoint LSPs for the specified instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

**Command Default**

No instance ID specified displays IS-IS checkpoint LSPs for all the IS-IS instances.

**Command Modes**

XR EXEC mode

**Command History**

Release 6.0 This command was introduced.

**Usage Guidelines**

The checkpointed LSPs displayed by this command are used to restore the LSP database during a Cisco-proprietary nonstop forwarding (NSF) restart. The `show isis checkpoint lsp` command, with the `show isis database` command, may be used to verify the consistency of the two databases.

**Examples**

The following is sample output from the `show isis checkpoint lsp` command:

```
RP/0/RP0/CPU0:router# 

show isis checkpoint lsp

Level  LSPID          Chkpt ID
       router-gsr6.00-00  80011f9c
       router-gsr6.01-00  80011f88
       router-gsr8.00-00  80011f74
       router-gsr9.00-00  80011f60
       router-gsr6.00-00  80011f4c
       router-gsr6.01-00  80011f38
       router-gsr8.00-00  80011f24
       router-gsr9.00-00  80011f10
Total LSP count:  8 (L1: 4, L2 4, local L1: 2, local L2 2)
```
This table describes the significant fields shown in the display.

Table 8: show isis checkpoint lsp Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>Routers with Level 1 or Level 2 adjacency configured.</td>
</tr>
<tr>
<td>LSPID</td>
<td>LSP identifier. The first six octets form the system ID of the router that originated the LSP. The next octet is the pseudonode ID. When this byte is 0 zero, the LSP describes links from the system. When it is nonzero, the LSP is a so-called nonpseudonode LSP. This is similar to a router link-state advertisement (LSA) in the Open Shortest Path First (OSPF) protocol. The LSP describes the state of the originating router. For each LAN, the designated router for that LAN creates and floods a pseudonode LSP, describing all systems attached to that LAN. The last octet is the LSP number. If there is more data than can fit in a single LSP, the LSP is divided into multiple LSP fragments. Each fragment has a different LSP number. An asterisk (*) indicates that the LSP was originated by the system on which this command is issued.</td>
</tr>
<tr>
<td>Chkpt ID</td>
<td>Unique ID issued to the checkpoint at its creation.</td>
</tr>
</tbody>
</table>
**show isis database**

To display the Intermediate System-to-Intermediate System (IS-IS) link-state packet (LSP) database, use the `show isis database` command in XR EXEC mode.

```
show isis [instance instance-id] database [level {1|2}] [update] [summary] [detail] [verbose] [lsp-id]
```

**Syntax Description**

- `instance instance-id` (Optional) Displays the IS-IS LSP database for the specified instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

- `level {1 | 2}` (Optional) Displays the IS-IS LSP database for Level 1 or Level 2 independently.

- `update` (Optional) Displays contents of LSP database managed by update thread.

- `summary` (Optional) Displays the LSP ID number, sequence number, checksum, hold time, and bit information.

- `detail` (Optional) Displays the contents of each LSP.

- `verbose` (Optional) Displays the contents of each LSP.

- `* | lsp-id` (Optional) LSP protocol data units (PDUs) identifier. Displays the contents of a single LSP by its ID number or may contain an `*` as a wildcard character.

**Command Default**

No instance ID specified displays the IS-IS LSP database for all the IS-IS instances.

Both Level 1 and Level 2 is configured if no level is specified.

**Command Modes**

XR EXEC mode

**Command History**

- **Release 6.0**  This command was introduced.
- **Release 6.5.1**  The output of this command shows an additional field called `Rcvd`.

**Usage Guidelines**

Each of the options for the `show isis database` command can be entered in an arbitrary string within the same command entry. For example, the following are both valid command specifications and provide the same output:  `show isis database detail level 2` and `show isis database level 2 detail`.

The `summary` keyword used with this command allows you to filter through a large IS-IS database and quickly identify problematic areas.

**Task ID**

- `isis`  `read`
Examples

The following is sample output from the `show isis database` command with no keywords specified:

```
RP/0/RP0/CPU0:router# show isis database

IS-IS Area a1 (Level-1) Link State Database

LSPID    LSP Seq Num  LSP Checksum  LSP Holdtime  ATT/P/OL
router-gsr6.00-00  * 0x00000016  0x62c8   896         0/0/0
router-gsr6.01-00  * 0x0000000f  0x56d9   902         0/0/0
router-gsr8.00-00  0x00000019  0x4b6d   1015        0/0/0
router-gsr9.00-00  0x00000016  0x33b7  957         0/0/0

Total LSP count: 4 (L1: 4, L2 0, local L1: 2, local L2 0)
```

The following sample output shows the remaining lifetime value that is received in LSP database. The received value is shown in the output under the `Rcvd` field.

```
RP/0/RP0/CPU0:ios#show isis database
Thu Dec 14 16:03:45.131 EST

IS-IS 100 (Level-1) Link State Database

LSPID    LSP Seq Num  LSP Checksum  LSP Holdtime  Rcvd  ATT/P/OL
1111.1111.1111.00-00  * 0x0073000e  0x0001  1002/*  1/0/1
1111.1111.1111.00-00  0x0073004a  0x0001   1195/59  1/0/1
1111.1111.1111.01-00  0x007300b8  0x0001   1178/59  0/0/1
1111.1111.1111.03-00  0x007300b6  0x0001   1179/59  0/0/1
1111.1111.1111.00-00  0x007300ad  0x0001   533/1200 1/0/1

Total Level-1 LSP count: 5  Local Level-1 LSP count: 1
```

This table describes the significant fields shown in the display.

**Table 9: show isis database Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSPID</td>
<td>LSP identifier. The first six octets form the system ID of the router that originated the LSP. The next octet is the pseudonode ID. When this byte is 0, the LSP describes links from the system. When it is nonzero, the LSP is a so-called nonpseudonode LSP. This is similar to a router link-state advertisement (LSA) in the Open Shortest Path First (OSPF) protocol. The LSP describes the state of the originating router. For each LAN, the designated router for that LAN creates and floods a pseudonode LSP, describing all systems attached to that LAN. The last octet is the LSP number. If there is more data than can fit in a single LSP, the LSP is divided into multiple LSP fragments. Each fragment has a different LSP number. An asterisk (*) indicates that the LSP was originated by the system on which this command is issued.</td>
</tr>
<tr>
<td>LSP Seq Num</td>
<td>Sequence number for the LSP that allows other systems to determine if they have received the latest information from the source.</td>
</tr>
<tr>
<td>LSP Checksum</td>
<td>Checksum of the entire LSP packet.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>LSP Holdtime</td>
<td>Time the LSP remains valid (in seconds). An LSP hold time of 0 indicates that this LSP was purged and is being removed from the link-state database (LSDB) of all routers. The value indicates how long the purged LSP stays in the LSDB before being completely removed.</td>
</tr>
</tbody>
</table>
| ATT/P/OL    | ATT—Attach bit. This bit indicates that the router is also a Level 2 router, and it can reach other areas. Level 1-only routers and Level 1-2 routers that have lost connection to other Level 2 routers use the Attach bit to find the closest Level 2 router. They point to a default route to the closest Level 2 router.  
P—P bit. Detects if the intermediate system is area partition repair capable. Cisco and other vendors do not support area partition repair.  
OL—Overload bit. Determines if the IS is congested. If the Overload bit is set, other routers do not use this system as a transit router when calculating routers. Only packets for destinations directly connected to the overloaded router are sent to this router. |

The following is sample output from the `show isis database` command with the `summary` keyword:

```
RP/0/RP0/CPU0# show isis database summary

IS-IS 10 Database Summary for all LSPs

<table>
<thead>
<tr>
<th></th>
<th>Active</th>
<th>Purged</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>L1</td>
<td>L2</td>
<td>Total</td>
</tr>
<tr>
<td>Fragment 0 Counts</td>
<td>-------</td>
<td>--------</td>
<td>-------</td>
</tr>
<tr>
<td>Router LSPs</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Pseudo-node LSPs</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>All LSPs</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Per Topology

IPv4 Unicast

<table>
<thead>
<tr>
<th></th>
<th>Active</th>
<th>Purged</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATT bit set LSPs</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>OVL bit set LSPs</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

All Fragment Counts

<table>
<thead>
<tr>
<th></th>
<th>Active</th>
<th>Purged</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router LSPs</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Pseudo-node LSPs</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>All LSPs</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>
```

This table describes the significant fields shown in the display.

**Table 10: show isis database summary Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router LSPs</td>
<td>Active, purged, and total LSPs associated with routers.</td>
</tr>
<tr>
<td>Pseudo-node LSPs</td>
<td>Active, purged, and total LSPs associated with pseudonodes.</td>
</tr>
<tr>
<td>All LSPs</td>
<td>Total active and purged LSPs.</td>
</tr>
<tr>
<td>ATT bit set LSPs</td>
<td>Attach bit (ATT). Indicates that the router is also a Level 2 router, and it can reach other areas. Level 1-only routers and Level 1-2 routers that have lost connection to other Level 2 routers use the Attach bit to find the closest Level 2 router. They point to a default route to the closest Level 2 router.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Overload bit. LSPs</td>
<td>Indicates if the IS is congested. If the Overload bit is set, other routers do not use this system as a transit router when calculating routers. Only packets for destinations directly connected to the overloaded router are sent to this router.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show isis database` command with the `detail verbose` keyword specified:

```
RP/R0/RP0/CPU0:router# show isis instance isp database detail verbose

IS-IS isp test (Level-1) Link State Database
LSPID        LSP Seq Num    LSP Checksum    LSP Holdtime  ATT/P/OL 0/0/0
router-5router1.00-00 0x00000003 * 0x00000d44 0x8074460 0x7e2c 457 535
Area Address: 4900
Area Address: 01
NLPID: 0xc0
NLPID: Hostname: router-50x8e
  IP Address: 172.1.1.5
  Metric: MT: 0 IP 172.3.55.0/24Standard (IPv4 Unicast)
  Metric: 10    IP 172.6.1.0/24
  MT:    IPv6 Unicast    Metric: 10 IP 172.7.0.0/24
  MT:    IPv4 Multicast  Metric: 10 IP 172.7.0.0/24
  Metric: 10 IS router-11.00
  Metric: 10 IS router-11.01
  MT:    IPv6 Multicast  router-11.00-00 * 0x0000000b 0x8074460
  Metric: 0 IS router-11.00
  Metric: 0 IS router-11.00
  MT:    IPv6 Multicast  router-11.00-00 * 0x0000000b 0x8074460
  Metric: 0 IS router-11.00
  Metric: 0 IS router-11.00
  Affinity: 0x00000000
  Interface IP Address: 10.3.11.145
  Neighbor IP Address: 10.3.11.143
  Physical BW: 155520 kbits/sec
  Total LSP count: 3 (L1: 3, L2 0, local L1: 2, local L2 0)
  Reservable Global pool BW: 0 kbits/sec
  Global Pool BW Unreserved:
  IS-IS isp (Level-2) Link State Database
LSPID        LSP Seq Num    LSP Checksum    LSP Holdtime  ATT/P/OL 0/0/0
router-5.00-00 0x00000005 0x807997c 457 0/0/0
  [0]: 0 kbits/sec    [1]: 0 kbits/sec
  [2]: 0 kbits/sec    [3]: 0 kbits/sec
  [4]: 0 kbits/sec    [5]: 0 kbits/sec
  [6]: 0 kbits/sec  [7]: 0 kbits/sec
  MPLS SRLG: Area Address: 49router2.00
  Interface IP Address: 10.3.11.145
  Neighbor IP Address: 10.3.11.143
  NLPID: 0xc0
  HostnameSRLGs: router-5IP Address[0]: 172.6.10,
  [1.5]: 20
  Metric: 0 10 IP 172IP-Extended 10.3.5511.0/24
```
As the output shows, besides the information displayed with the `show isis database` command, the command with the `detail verbose` keyword displays the contents of each LSP.

Table 11: `show isis instance isp database detail` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSPID</td>
<td>LSP identifier. The first six octets form the system ID of the router that originated the LSP. The next octet is the pseudonode ID. When this byte is 0, the LSP describes links from the system. When it is nonzero, the LSP is a so-called nonpseudonode LSP. This is similar to a router link-state advertisement (LSA) in the Open Shortest Path First (OSPF) protocol. The LSP describes the state of the originating router. For each LAN, the designated router for that LAN creates and floods a pseudonode LSP, describing all systems attached to that LAN. The last octet is the LSP number. If there is more data than can fit in a single LSP, the LSP is divided into multiple LSP fragments. Each fragment has a different LSP number. An asterisk (*) indicates that the LSP was originated by the system on which this command is issued.</td>
</tr>
<tr>
<td>LSP Seq Num</td>
<td>Sequence number for the LSP that allows other systems to determine if they have received the latest information from the source.</td>
</tr>
<tr>
<td>LSP Checksum</td>
<td>Checksum of the entire LSP packet.</td>
</tr>
<tr>
<td>LSP Holdtime</td>
<td>Amount of time the LSP remains valid (in seconds). An LSP hold time of 0 indicates that this LSP was purged and is being removed from the link-state database (LSDB) of all routers. The value indicates how long the purged LSP stays in the LSDB before being completely removed.</td>
</tr>
</tbody>
</table>
**ATT/P/OL**
- **ATT**—Attach bit. This bit indicates that the router is also a Level 2 router, and it can reach other areas. Level 1-only routers and Level 1-2 routers that have lost connection to other Level 2 routers use the Attach bit to find the closest Level 2 router. They point to a default route to the closest Level 2 router.
- **P**—P bit. Detects if the intermediate system is area partition repair capable. Cisco and other vendors do not support area partition repair.
- **OL**—Overload bit. Determines if the IS is congested. If the Overload bit is set, other routers do not use this system as a transit router when calculating routers. Only packets for destinations directly connected to the overloaded router are sent to this router.

**Area Address**
- Reachable area addresses from the router. For Level 1 LSPs, these are the area addresses configured manually on the originating router. For Level 2 LSPs, these are all the area addresses for the area this route belongs to.

**NLPID**
- Network Layer Protocol Identifier.

**Hostname**
- Hostname of the node.

**IP Address**
- Address of the node.

**Metric**
- IS-IS metric for the cost of the adjacency between the originating router and the advertised neighbor, or the metric of the cost to get from the advertising router to the advertised destination (which can be an IP address, an end system (ES), or a Connectionless Network Service (CLNS) prefix).

The following is additional sample output from the `show isis database detail` command. This is a Level 2 LSP. The area address 39.0001 is the address of the area in which the router resides.

```
RP/0/RSP0/CPU0:router# show isis database level 2 detail
IS-IS Level-2 Link State Database
LSPID LSP Seq Num LSP Checksum LSP Holdtime ATT/P/OL
0000.0C00.1111.00-00* 0x00000006 0x4DB3 1194 0/0/0
  Area Address: 39.0001
  NLPID: 0x81 0xCC
  IP Address: 172.18.1.17
  Metric: 10 IS 0000.0C00.1111.09
  Metric: 10 IS 0000.0C00.1111.08
  Metric: 10 IP 172.17.4.0 255.255.255.0
  Metric: 10 IP 172.18.8.0 255.255.255.0
  Metric: 0 IP-External 10.0.0.0 255.0.0.0
```

The IP entries are the directly connected IP subnets the router is advertising (with associated metrics). The IP-External entry is a redistributed route.
Table 12: show isis database level 2 detail Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSPID</td>
<td>LSP identifier. The first six octets form the system ID of the router that originated the LSP. The next octet is the pseudonode ID. When this byte is 0, the LSP describes links from the system. When it is nonzero, the LSP is a so-called nonpseudonode LSP. This is similar to a router link-state advertisement (LSA) in the Open Shortest Path First (OSPF) protocol. The LSP describes the state of the originating router. For each LAN, the designated router for that LAN creates and floods a pseudonode LSP, describing all systems attached to that LAN. The last octet is the LSP number. If there is more data than can fit in a single LSP, the LSP is divided into multiple LSP fragments. Each fragment has a different LSP number. An asterisk (*) indicates that the LSP was originated by the system on which this command is issued.</td>
</tr>
<tr>
<td>LSP Seq Num</td>
<td>Sequence number for the LSP that allows other systems to determine if they have received the latest information from the source.</td>
</tr>
<tr>
<td>LSP Checksum</td>
<td>Checksum of the entire LSP packet.</td>
</tr>
<tr>
<td>LSP Holdtime</td>
<td>Time the LSP remains valid (in seconds). An LSP hold time of 0 indicates that this LSP was purged and is being removed from the link-state database (LSDB) of all routers. The value indicates how long the purged LSP stays in the LSDB before being completely removed.</td>
</tr>
<tr>
<td>ATT/P/OL</td>
<td>ATT—Attach bit. This bit indicates that the router is also a Level 2 router, and it can reach other areas. Level 1-only routers and Level 1-2 routers that have lost connection to other Level 2 routers use the Attach bit to find the closest Level 2 router. They point to a default route to the closest Level 2 router. P—P bit. Detects if the intermediate system is area partition repair capable. Cisco and other vendors do not support area partition repair. OL—Overload bit. Determines if the IS is congested. If the Overload bit is set, other routers do not use this system as a transit router when calculating routers. Only packets for destinations directly connected to the overloaded router are sent to this router.</td>
</tr>
<tr>
<td>Area Address</td>
<td>Reachable area addresses from the router. For Level 1 LSPs, these are the area addresses configured manually on the originating router. For Level 2 LSPs, these are all the area addresses for the area to which this route belongs.</td>
</tr>
<tr>
<td>NLPID</td>
<td>Network Layer Protocol Identifier.</td>
</tr>
<tr>
<td>Hostname</td>
<td>Hostname of the node.</td>
</tr>
<tr>
<td>IP Address:</td>
<td>IP address of the node.</td>
</tr>
<tr>
<td>Metric:</td>
<td>IS-IS metric for the cost of the adjacency between the originating router and the advertised neighbor, or the metric of the cost to get from the advertising router to the advertised destination (which can be an IP address, an end system (ES), or a Connectionless Network Service (CLNS) prefix).</td>
</tr>
</tbody>
</table>
Table 13: show isis database verbose Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSPID</td>
<td>LSP identifier. The first six octets form the system ID of the router that originated the LSP.</td>
</tr>
<tr>
<td></td>
<td>The next octet is the pseudonode ID. When this byte is zero, the LSP describes links from the system. When it is nonzero, the LSP is a so-called nonpseudonode LSP. This is similar to a router link-state advertisement (LSA) in the Open Shortest Path First (OSPF) protocol. The LSP describes the state of the originating router.</td>
</tr>
<tr>
<td></td>
<td>For each LAN, the designated router for that LAN creates and floods a pseudonode LSP, describing all systems attached to that LAN.</td>
</tr>
<tr>
<td></td>
<td>The last octet is the LSP number. If there is more data than can fit in a single LSP, the LSP is divided into multiple LSP fragments. Each fragment has a different LSP number. An asterisk (*) indicates that the LSP was originated by the system on which this command is issued.</td>
</tr>
<tr>
<td>LSP Seq Num</td>
<td>Sequence number for the LSP that allows other systems to determine if they have received the latest information from the source.</td>
</tr>
<tr>
<td>LSP Checksum</td>
<td>Checksum of the entire LSP packet.</td>
</tr>
<tr>
<td>LSP Holdtime</td>
<td>Time the LSP remains valid (in seconds). An LSP hold time of zero indicates that this LSP was purged and is being removed from the link-state database (LSDB) of all routers. The value indicates how long the purged LSP stays in the LSDB before being completely removed.</td>
</tr>
<tr>
<td>ATT/P/OL</td>
<td>ATT—Attach bit. This bit indicates that the router is also a Level 2 router, and it can reach other areas. Level 1-only routers and Level 1-2 routers that have lost connection to other Level 2 routers use the Attach bit to find the closest Level 2 router. They point to a default route to the closest Level 2 router.</td>
</tr>
<tr>
<td></td>
<td>P—P bit. Detects if the intermediate system is area partition repair capable. Cisco and other vendors do not support area partition repair.</td>
</tr>
<tr>
<td></td>
<td>OL—Overload bit. Determines if the IS is congested. If the Overload bit is set, other routers do not use this system as a transit router when calculating routers. Only packets for destinations directly connected to the overloaded router are sent to this router.</td>
</tr>
<tr>
<td>Area Address</td>
<td>Reachable area addresses from the router. For Level 1 LSPs, these are the area addresses configured manually on the originating router. For Level 2 LSPs, these are all the area addresses for the area to which this route belongs.</td>
</tr>
<tr>
<td>NLPI D</td>
<td>Network Layer Protocol Identifier.</td>
</tr>
<tr>
<td>Hostname</td>
<td>Hostname of the node.</td>
</tr>
<tr>
<td>IP Address</td>
<td>IP address of the node.</td>
</tr>
<tr>
<td>Metric</td>
<td>IS-IS metric for the cost of the adjacency between the originating router and the advertised neighbor, or the metric of the cost to get from the advertising router to the advertised destination (which can be an IP address, an end system (ES), or a Connectionless Network Service (CLNS) prefix).</td>
</tr>
</tbody>
</table>
### Table: IS-IS TLV Information

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPLS SRLG</td>
<td>MPLS SRLG TLV information per neighbor, identified by hostname or system ID.</td>
</tr>
<tr>
<td>Interface IP Address</td>
<td>Local interface IP address.</td>
</tr>
<tr>
<td>Neighbor IP Address</td>
<td>Remote interface IP address.</td>
</tr>
<tr>
<td>Flags</td>
<td>Flags carried in SRLG TLV. The Least Significant Bit (LSB) is set if the interface is numbered.</td>
</tr>
<tr>
<td>SRLGs</td>
<td>SRLG values.</td>
</tr>
</tbody>
</table>

This is the sample output from the `show isis database verbose` command. The output shows IPv4 adjacency segment ID (SID), prefix (node) SID, and Segment Routing Global Block (SRGB) values.

```
RP/0/RP0/CPU0:router# show isis database verbose
Fri May 2 17:53:44.575 PDT
IS-IS DEFAULT (Level-1) Link State Database
LSPID LSP Seq Num LSP Checksum LSP Holdtime ATT/P/OL
p1.00-00 0x00000080 0x4780 1044 1/0/0
Area Address: 49.0001
NLPID: 0xcc
NLPID: 0x8e
MT: Standard (IPv4 Unicast)
MT: IPv6 Unicast 1/0/0
Hostname: p1
IP Address: 172.16.255.101
IPv6 Address: 2001:db8::ff:101
Router Cap: 172.16.255.101, D:0, S:0
  Segment Routing: I:1 V:0, SRGB Base: 16000 Range: 7999
  Metric: 10 IS-Extended p2.00
  Interface IP Address: 172.16.2.4
  Neighbor IP Address: 172.16.2.5
  ADJ-SID: F:0 B:0 weight:0 Adjacency-sid:24002
  Metric: 10 IS-Extended p2.00
  Interface IP Address: 172.16.2.4
  Neighbor IP Address: 172.16.2.5
  ADJ-SID: F:0 B:0 weight:0 Adjacency-sid:24003
  Metric: 10 IP-Extended 172.16.1.0/31
  Metric: 10 IP-Extended 172.16.2.2/31
  Metric: 10 IP-Extended 172.16.2.4/31
  Metric: 10 IP-Extended Interarea 172.16.255.2/32
  Admin. Tag: 255
  Prefix-SID Index: 42, R:1 N:0 P:1
  Metric: 0 IP-Extended 172.16.255.101/32
    Prefix-SID Index: 141, R:0 N:0 P:0
  Metric: 10 MT (IPv6 Unicast) IS-Extended p2.00
  Metric: 10 MT (IPv6 Unicast) IS-Extended p2.00
  Metric: 10 MT (IPv6 Unicast) IPv6 2001:db8::1:0/127
  Metric: 10 MT (IPv6 Unicast) IPv6 2001:db8::2:2/127
  Metric: 10 MT (IPv6 Unicast) IPv6 2001:db8::2:4/127
  Metric: 10 MT (IPv6 Unicast) IPv6-Interarea 2001:db8::ff:2/128
  Admin. Tag: 255
  Metric: 0 MT (IPv6 Unicast) IPv6 2001:db8::ff:101/128
```
show isis database-log

To display the entries in the Intermediate System-to-Intermediate System (IS-IS) database log, use the show isis database-log command in XR EXEC mode.

```
show isis database-log [level {1|2}] [last number|first number]
```

**Syntax Description**

- `level {1 | 2}` (Optional) Displays the database log for Level 1 or Level 2 independently.
- `last number` (Optional) Specifies that the output be restricted to the last `number` of entries. Range is 1 to 1000.
- `first number` (Optional) Specifies that the output be restricted to the first `number` of entries. Range is 1 to 1000.

**Command Default**

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

```
Task ID
Operations ID
isis  read
```

**Examples**

The following is sample output from the `show isis database-log` command:

```
RP/0/RP0/CPU0# show isis database-log

IS-IS 10 Level 1 Link State Database Log

<table>
<thead>
<tr>
<th>WHEN</th>
<th>LSPID</th>
<th>New LSP Op</th>
<th>Seq Num</th>
<th>Holdtime OL</th>
<th>Old LSP Seq Num</th>
<th>Holdtime OL</th>
</tr>
</thead>
<tbody>
<tr>
<td>01:17:19</td>
<td>12b1.03-00</td>
<td>REP 0x00000003 1200</td>
<td>0x00000002</td>
<td>340</td>
<td>0x00000000</td>
<td>0</td>
</tr>
<tr>
<td>00:16:20</td>
<td>12b1.00-00</td>
<td>REP 0x000001d8 1200</td>
<td>0x000001d7</td>
<td>375</td>
<td>0x000001d7</td>
<td>375</td>
</tr>
<tr>
<td>01:06:00</td>
<td>12b1.03-00</td>
<td>REP 0x00000004 1200</td>
<td>0x00000003</td>
<td>520</td>
<td>0x00000000</td>
<td>0</td>
</tr>
<tr>
<td>01:05:46</td>
<td>12a1.00-00</td>
<td>REP 0x00000001fc 1200</td>
<td>0x00000001fb</td>
<td>425</td>
<td>0x00000001fb</td>
<td>425</td>
</tr>
<tr>
<td>00:55:01</td>
<td>12b1.00-00</td>
<td>REP 0x0000001d9 1200</td>
<td>0x0000001d8</td>
<td>520</td>
<td>0x0000001d8</td>
<td>520</td>
</tr>
<tr>
<td>00:53:39</td>
<td>12b1.03-00</td>
<td>REP 0x0000002005 1200</td>
<td>0x0000002004</td>
<td>459</td>
<td>0x0000002004</td>
<td>459</td>
</tr>
<tr>
<td>00:53:19</td>
<td>12a1.00-00</td>
<td>REP 0x00000001fd 1200</td>
<td>0x00000001fc</td>
<td>453</td>
<td>0x00000001fc</td>
<td>453</td>
</tr>
<tr>
<td>00:42:12</td>
<td>12b1.00-00</td>
<td>REP 0x0000001da 1200</td>
<td>0x0000001d9</td>
<td>431</td>
<td>0x0000001d9</td>
<td>431</td>
</tr>
<tr>
<td>00:39:56</td>
<td>12b1.03-00</td>
<td>REP 0x0000001d2 1200</td>
<td>0x0000001d1</td>
<td>430</td>
<td>0x0000001d1</td>
<td>430</td>
</tr>
<tr>
<td>00:38:54</td>
<td>12a1.00-00</td>
<td>REP 0x0000001fe 1200</td>
<td>0x0000001fd</td>
<td>334</td>
<td>0x0000001fd</td>
<td>334</td>
</tr>
<tr>
<td>00:29:10</td>
<td>12b1.00-00</td>
<td>REP 0x0000001db 1200</td>
<td>0x0000001da</td>
<td>418</td>
<td>0x0000001da</td>
<td>418</td>
</tr>
<tr>
<td>00:27:22</td>
<td>12b1.03-00</td>
<td>REP 0x000000007 1200</td>
<td>0x000000006</td>
<td>446</td>
<td>0x000000006</td>
<td>446</td>
</tr>
<tr>
<td>00:25:10</td>
<td>12a1.00-00</td>
<td>REP 0x0000001ff 1200</td>
<td>0x0000001fe</td>
<td>375</td>
<td>0x0000001fe</td>
<td>375</td>
</tr>
<tr>
<td>00:17:04</td>
<td>12b1.00-00</td>
<td>REP 0x0000001dc 1200</td>
<td>0x00000001db</td>
<td>473</td>
<td>0x00000001db</td>
<td>473</td>
</tr>
</tbody>
</table>
```
This table describes the significant fields shown in the display.

**Table 14: show isis database-log Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>WHEN</td>
<td>Elapsed time (in hh:mm:ss) since the event was logged.</td>
</tr>
<tr>
<td>LSPID</td>
<td>LSP identifier. The first six octets form the system ID of the router that originated the LSP. The next octet is the pseudonode ID. When this byte is 0 zero, the LSP describes links from the system. When it is nonzero, the LSP is a so-called nonpseudonode LSP. This is similar to a router link-state advertisement (LSA) in the Open Shortest Path First (OSPF) protocol. The LSP describes the state of the originating router. For each LAN, the designated router for that LAN creates and floods a pseudonode LSP, describing all systems attached to that LAN. The last octet is the LSP number. If there is more data than can fit in a single LSP, the LSP is divided into multiple LSP fragments. Each fragment has a different LSP number. An asterisk (*) indicates that the LSP was originated by the system on which this command is issued.</td>
</tr>
<tr>
<td>New LSP</td>
<td>New router or pseudonode appearing in the topology.</td>
</tr>
<tr>
<td>Old LSP</td>
<td>Old router or pseudonode leaving the topology.</td>
</tr>
<tr>
<td>Op</td>
<td>Operation on the database: inserted (INS) or replaced (REP).</td>
</tr>
<tr>
<td>Seq Num</td>
<td>Sequence number for the LSP that allows other systems to determine if they have received the latest information from the source.</td>
</tr>
<tr>
<td>Holdtime</td>
<td>Time the LSP remains valid (in seconds). An LSP hold time of 0 indicates that this LSP was purged and is being removed from the link-state database (LSDB) of all routers. The value indicates how long the purged LSP stays in the LSDB before being completely removed.</td>
</tr>
<tr>
<td>OL</td>
<td>Overload bit. Determines if the IS is congested. If the Overload bit is set, other routers do not use this system as a transit router when calculating routers. Only packets for destinations directly connected to the overloaded router are sent to this router.</td>
</tr>
</tbody>
</table>
show isis hostname

To display the entries in the Intermediate System-to-Intermediate System (IS-IS) router name-to-system ID mapping table, use the `show isis hostname` command in XR EXEC mode.

```
show isis [instance instance-id] hostname
```

**Syntax Description**

- `instance instance-id` (Optional) Displays the IS-IS router name-to-system ID mapping table for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

**Command Default**

No instance ID specified displays the IS-IS router name-to-system ID mapping table for all the IS-IS instances.

**Command Modes**

XR EXEC mode

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

The `show isis hostname` command does not display entries if the dynamic hostnames are disabled.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show isis hostname` command with the `instance` and `instance-id` values specified:

```
RP/0/RP0/CPU0:router# show isis instance isp hostname

ISIS isp hostnames
Level  System ID   Dynamic Hostname
1  0001.0000.0005 router
2  * 0001.0000.0011 router-11
```

This table describes the significant fields shown in the display.

**Table 15: show isis instance isp hostname Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>IS-IS level of the router.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>System ID</td>
<td>Dynamic hostname of the system. The hostname is specified using the <code>hostname</code> command. If the dynamic hostname is not known or <code>hostname dynamic disable</code> command has been executed, the 6-octet system ID is used.</td>
</tr>
<tr>
<td>Dynamic Hostname</td>
<td>Hostname of the router.</td>
</tr>
<tr>
<td>*</td>
<td>Local router.</td>
</tr>
</tbody>
</table>
show isis interface

To display information about the Intermediate System-to-Intermediate System (IS-IS) interfaces, use the
show isis interface command in XR EXEC mode.

show isis interface [type interface-path-id]level {1|2} [brief]

Syntax Description

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Interface type. For more information, use the question mark (?) online help function.</td>
</tr>
<tr>
<td>interface-path-id</td>
<td>Physical interface or virtual interface.</td>
</tr>
<tr>
<td>Note</td>
<td>Use the show interfaces command to see a list of all interfaces currently configured on the router.</td>
</tr>
<tr>
<td>For more information about the syntax for the router, use the question mark (?) online help function.</td>
<td></td>
</tr>
<tr>
<td>level {1</td>
<td>2} (Optional) Displays IS-IS interface information for Level 1 or Level 2 independently.</td>
</tr>
<tr>
<td>brief (Optional) Displays brief interface output.</td>
<td></td>
</tr>
</tbody>
</table>

Command Default

Displays all IS-IS interfaces.

Command Modes

XR EXEC mode

Command History

Release 6.0 This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the show isis interface command:

RP/RP0/CPU0:router#show isis interface

    tenGigE 0/3/0/2
    tenGigE 0/3/0/2   Enabled
    Adjacency Formation:   Enabled
    Prefix Advertisement:   Enabled
    BFD:   Disabled
    BFD Min Interval:  150
    BFD Multiplier:  3
    Circuit Type:   level-2-only
    Media Type:  P2P
    Circuit Number:  0
Extended Circuit Number: 67111168
Next P2P IIH in: 4 s
LSP Rexmit Queue Size: 0

Level-2
Adjacency Count: 1
LSP Pacing Interval: 33 ms
PSNP Entry Queue Size: 0

CLNS I/O
Protocol State: Up
MTU: 4469

IPv4 Unicast Topology: Enabled
Adjacency Formation: Running
Prefix Advertisement: Running
Metric (L1/L2): 10/100
MPLS LDP Sync (L1/L2): Disabled/Disabled
IPv6 Unicast Topology: Disabled (Not cfg on the intf)

IPv4 Address Family: Enabled
Protocol State: Up
Forwarding Address(es): 10.3.10.143
Global Prefix(es): 10.3.10.0/24
IPv6 Address Family: Disabled (No topology enabled which uses IPv6)

LSP transmit timer expires in 0 ms
LSP transmission is idle
Can send up to 9 back-to-back LSPs in the next 0 ms

This table describes the significant fields shown in the display.

Table 16: show isis interface Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tenGigE0/6/0/0</td>
<td>Status of the interface, either enabled or disabled.</td>
</tr>
<tr>
<td>Adjacency formation:</td>
<td>Status of adjacency formation, either enabled or disabled.</td>
</tr>
<tr>
<td>Prefix Advertisement:</td>
<td>Status of advertising connected prefixes, either enabled or disabled.</td>
</tr>
<tr>
<td>BFD:</td>
<td>Status of Bidirectional Forwarding Detection (BFD), either enabled or disabled.</td>
</tr>
<tr>
<td>BFD Min Interval:</td>
<td>BFD minimum interval.</td>
</tr>
<tr>
<td>BFD Multiplier:</td>
<td>BFD multiplier.</td>
</tr>
<tr>
<td>Circuit Type:</td>
<td>Levels the interface is running on (circuit-type configuration) which may be a subset of levels on the router.</td>
</tr>
<tr>
<td>Media Type:</td>
<td>Media type on which IS-IS is running.</td>
</tr>
<tr>
<td>Circuit Number:</td>
<td>Unique ID assigned to a circuit internally (8-bit integer).</td>
</tr>
<tr>
<td>Extended Circuit Number:</td>
<td>Valid only for point-to-point interfaces (32-bit integer).</td>
</tr>
<tr>
<td>LSP Rexmit Queue Size:</td>
<td>Number of LSPs pending retransmission on the interface.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Adjacency Count</td>
<td>Number of adjacencies formed with a neighboring router that supports the same set of protocols.</td>
</tr>
<tr>
<td>PSNP Entry Queue Size</td>
<td>Number of SNP entries pending inclusion in the next PSNP.</td>
</tr>
<tr>
<td>LAN ID</td>
<td>ID of the LAN.</td>
</tr>
<tr>
<td>Priority (Local/DIS)</td>
<td>Priority of this interface or priority of the Designated Intermediate System.</td>
</tr>
<tr>
<td>Next LAN IIH in</td>
<td>Time (in seconds) in which the next LAN hello message is sent.</td>
</tr>
<tr>
<td>LSP Pacing Interval</td>
<td>Interval at which the link-state packet (LSP) transmission rate (and by implication the reception rate of other systems) is to be reduced.</td>
</tr>
<tr>
<td>Protocol State</td>
<td>Running state of the protocol (up or down).</td>
</tr>
<tr>
<td>MTU</td>
<td>Link maximum transmission unit (MTU).</td>
</tr>
<tr>
<td>SNPA</td>
<td>Data-link address (also known as the Subnetwork Point of Attachment [SNPA]) of the neighbor.</td>
</tr>
<tr>
<td>All Level-n ISs</td>
<td>Status of interface membership in Layer 2 multicast group. The status options are Yes or reason for not being a member of the multicast group.</td>
</tr>
<tr>
<td>IPv4 Unicast Topology</td>
<td>Status of the topology, either enabled or disabled.</td>
</tr>
<tr>
<td>Adjacency Formation</td>
<td>Status of adjacency formation. The status options are Running or a reason for not being ready to form adjacencies.</td>
</tr>
<tr>
<td>Prefix Advertisement</td>
<td>Status of advertising prefixes, either enabled or disabled.</td>
</tr>
<tr>
<td>Metric (L1/L2)</td>
<td>IS-IS metric for the cost of the adjacency between the originating router and the advertised neighbor, or the metric of the cost to get from the advertising router to the advertised destination (which can be an IP address, an end system (ES), or a connectionless network service (CLNS) prefix).</td>
</tr>
<tr>
<td>MPLS LDP Sync (L1/L2)</td>
<td>Status of LDP IS-IS synchronization, either enabled or disabled. When enabled, the state of synchronization (Sync Status) is additionally displayed as either achieved or not achieved.</td>
</tr>
<tr>
<td>IPv4 Address Family</td>
<td>Status of the address family, either enabled or disabled.</td>
</tr>
<tr>
<td>Protocol State</td>
<td>State of the protocol.</td>
</tr>
<tr>
<td>Forwarding Address(es)</td>
<td>Addresses on this interface used by the neighbor for next-hop forwarding.</td>
</tr>
<tr>
<td>Global Prefix(es)</td>
<td>Prefixes for this interface included in the LSP.</td>
</tr>
<tr>
<td>LSP transmit timer expires in</td>
<td>LSP transmission expiration timer interval (in milliseconds).</td>
</tr>
</tbody>
</table>
### Description Field

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSP transmission is</td>
<td>State of LSP transmission. Valid states are:</td>
</tr>
<tr>
<td></td>
<td>• idle</td>
</tr>
<tr>
<td></td>
<td>• in progress</td>
</tr>
<tr>
<td></td>
<td>• requested</td>
</tr>
<tr>
<td></td>
<td>• requested and in progress</td>
</tr>
</tbody>
</table>

The following is sample output from the `show isis interface` command with the `brief` keyword:

```
RP/0/0/CP00:router# show isis interface brief

<table>
<thead>
<tr>
<th>Interface</th>
<th>All Adjs</th>
<th>Adj Topos Run/Cfg</th>
<th>Adv Topos Run/Cfg</th>
<th>CLNS</th>
<th>MTU</th>
<th>Prio</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>OK</td>
<td>L1 L2</td>
<td>Run/Cfg</td>
<td>Run/Cfg</td>
<td></td>
<td>L1 L2</td>
</tr>
<tr>
<td>PO0/5/0/0</td>
<td>Yes</td>
<td>1</td>
<td>1/1</td>
<td>1/1</td>
<td>Up</td>
<td>4469</td>
</tr>
<tr>
<td>Gi0/6/0/0</td>
<td>Yes</td>
<td>1*</td>
<td>1*</td>
<td>1/1</td>
<td>Up</td>
<td>1497</td>
</tr>
</tbody>
</table>
```

This table describes the significant fields shown in the display.

*Table 17: show isis interface brief Field Descriptions*

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>Name of the interface.</td>
</tr>
<tr>
<td>All OK</td>
<td>Everything is working as expected for this interface.</td>
</tr>
<tr>
<td>Adjs L1 L2</td>
<td>Number of L1 and L2 adjacencies over this interface.</td>
</tr>
<tr>
<td>Adj Topos Run/Cfg</td>
<td>Number of topologies that participate in forming adjacencies. Number of topologies that were configured to participate in forming adjacencies.</td>
</tr>
<tr>
<td>Adv Topos Run/Cfg</td>
<td>Number of topologies that participate in advertising prefixes. Number of topologies that were configured to participate in advertising prefixes.</td>
</tr>
<tr>
<td>CLNS</td>
<td>Status of the Connectionless Network Service. Status options are Up or Down.</td>
</tr>
<tr>
<td>MTU</td>
<td>Maximum transfer unit size for the interface.</td>
</tr>
<tr>
<td>Prio L1 L2</td>
<td>Interface L1 priority. Interface L2 priority.</td>
</tr>
</tbody>
</table>
show isis lsp-log

To display link-state packet (LSP) log information, use the `show isis lsp-log` command in XR EXEC mode.

```
show isis [instance instance-id] lsp-log [level {1|2}] [{last number|first number}]
```

**Syntax Description**

- `instance instance-id` (Optional) Displays the LSP log information for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

- `level {1 | 2}` (Optional) Displays the Intermediate System-to-Intermediate System (IS-IS) link-state database for Level 1 or Level 2 independently.

- `last number` (Optional) Specifies that the output be restricted to the last `number` of entries. Range is 1 to 20.

- `first number` (Optional) Specifies that the output be restricted to the first `number` of entries. Range is 1 to 20.

**Command Default**

No instance ID specified displays the LSP log information for all the IS-IS instances.

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

XR EXEC mode

**Command History**

- **Release** 6.0 This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- **Task** isis
  - **Operations** read

**Examples**

The following is sample output from the `show isis lsp-log` command with the `instance` and `instance-id` values specified:

```
RP/0/RP0/CPU0:router# show isis instance isp lsp-log
ISIS isp Level 1 LSP log
When Count Interface Triggers
00:02:36 1
00:02:31 1 LSPREGEN
00:02:26 1 PO4/1 DELADJ
00:02:24 1 PO4/1 NEMADJ
00:02:23 1 Gi5/0 DIS
00:01:27 1 Lo0 IPDOWN
```
This table describes the significant fields shown in the display.

Table 18: show isis instance isp Lsp-log Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>IS-IS level of the router.</td>
</tr>
<tr>
<td>When</td>
<td>How long ago (hh:mm:ss) an LSP rebuild occurred. The last 20 occurrences are logged.</td>
</tr>
<tr>
<td>Count</td>
<td>Number of events that triggered this LSP run. When there is a topology change, often multiple LSPs are received in a short period. A router waits 5 seconds before running a full LSP, so it can include all new information. This count denotes the number of events (such as receiving new LSPs) that occurred while the router was waiting its 5 seconds before running full LSP.</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface that corresponds to the triggered reasons for the LSP rebuild.</td>
</tr>
<tr>
<td>Triggers</td>
<td>A list of all reasons that triggered an LSP rebuild. The triggers are:</td>
</tr>
<tr>
<td></td>
<td>• AREASET—area set changed</td>
</tr>
<tr>
<td></td>
<td>• ATTACHFLAG—bit attached</td>
</tr>
<tr>
<td></td>
<td>• CLEAR—clear command</td>
</tr>
<tr>
<td></td>
<td>• CONFIG—configuration change</td>
</tr>
<tr>
<td></td>
<td>• DELADJ—adjacency deleted</td>
</tr>
<tr>
<td></td>
<td>• DIS—DIS changed</td>
</tr>
<tr>
<td></td>
<td>• IFDOWN—interface down</td>
</tr>
<tr>
<td></td>
<td>• IPADDRCHG—IP address change</td>
</tr>
<tr>
<td></td>
<td>• IPDEFORIG—IP def-orig</td>
</tr>
<tr>
<td></td>
<td>• IPDOWN—connected IP down</td>
</tr>
<tr>
<td></td>
<td>• IFDOWN—interface down</td>
</tr>
<tr>
<td></td>
<td>• IPEXT—external IP</td>
</tr>
<tr>
<td></td>
<td>• IPIA—interarea IP</td>
</tr>
<tr>
<td></td>
<td>• IPUP—connected IP up</td>
</tr>
<tr>
<td></td>
<td>• LSPDBOL—LSPDBOL bit</td>
</tr>
<tr>
<td></td>
<td>• LSPREGEN—LSP regeneration</td>
</tr>
<tr>
<td></td>
<td>• NEWADJ—new adjacency</td>
</tr>
</tbody>
</table>
show isis mesh-group

To display Intermediate System-to-Intermediate System (IS-IS) mesh group information, use the show isis mesh-group command in XR EXEC mode.

```
show isis [instance instance-id] mesh-group
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance instance-id</td>
<td>(Optional) Displays the mesh group information for the specified IS-IS instance only.</td>
</tr>
</tbody>
</table>

- The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

**Command Default**

No instance ID specified displays the IS-IS mesh group information for all the IS-IS instances.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show isis mesh-group` command with the `instance` and `instance-id` values specified:

```
RP/0/RP0/CPU0:router# show isis instance isp mesh-group

ISIS isp Mesh Groups

Mesh group 6:
tenGigE 0/4/0/1
```
This table describes the significant fields shown in the display.

**Table 19: show isis instance isp mesh-group Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mesh group</td>
<td>Mesh group number to which this interface is a member. A mesh group optimizes link-state packet (LSP) flooding in nonbroadcast multiaccess (NBMA) networks with highly meshed, point-to-point topologies. LSPs that are first received on interfaces that are part of a mesh group are flooded to all interfaces except those in the same mesh group.</td>
</tr>
<tr>
<td>GigabitEthernet0/4/0/1</td>
<td>Interface belonging to mesh group 6.</td>
</tr>
</tbody>
</table>
show isis neighbors

To display information about Intermediate System-to-Intermediate System (IS-IS) neighbors, use the **show isis neighbors** command in XR EXEC mode.

```
show isis [instance instance-id] neighbors [type interface-path-id|summary] [detail] [systemid system-id]
```

**Syntax Description**

- **instance instance-id** (Optional) Displays the IS-IS neighbor information for the specified IS-IS instance only.
  - The *instance-id* argument is the instance identifier (alphanumeric) defined by the **router isis** command.
- **type** Interface type. For more information, use the question mark (?) online help function.
  - **Physical interface or virtual interface.**
- **interface-path-id** Physical interface or virtual interface.
  - **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.
  - For more information about the syntax for the router, use the question mark (?) online help function.
- **summary** (Optional) Displays neighbor status count for each level.
- **detail** (Optional) Displays additional details.
- **systemid system-id** (Optional) Displays the information for the specified neighbor only.

**Command Default**

No instance ID specified displays neighbor information for all the IS-IS instances.
Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

XR EXEC mode

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- **Task ID Operations ID**
  - **isis read**

**Examples**

The following is sample output from the **show isis neighbors** command with the **instance** and **instance-id** values specified:

Routing Command Reference for Cisco NCS 5000 Series Routers
Total neighbor count: 3
RP/0/RSP0/RP0/CP0# show isis instance isp neighbors detail

IS-IS isp neighbors:
- System Id: e222e, Interface: Gi0/1/0/0, SNPA: *PtoP*, State: Up, Holdtime: 23, Type: L1 Capable
  - Area Address(es): 00
  - IPv4 Address(es): 10.1.0.45*
  - IPv6 Address(es): fe80::212:daff:fe6b:68a8*
  - Topologies: 'IPv4 Unicast', 'IPv6 Unicast'
  - Uptime: 01:09:44
  - IPFRR: LFA Neighbor: elise
    - LFA IPv4 address: 10.100.1.2
    - LFA Router address: 192.168.0.45
- System Id: e333e, Interface: Gi0/1/0/0.1, SNPA: 0012.da6b.68a8, State: Up, Holdtime: 8, Type: L1 Capable
  - Area Address(es): 00
  - IPv4 Address(es): 10.100.1.2*
  - IPv6 Address(es): fe80::212:daff:fe62:e0a8*
  - Topologies: 'IPv4 Unicast'
  - Uptime: 01:09:46
  - IPFRR: LFA Neighbor: elise
    - LFA IPv4 address: 10.10.0.45
    - LFA Router address: 192.168.0.45
    - LFA Interface: Gi0/1/0/0
- System Id: m44i, Interface: Gi0/1/0/1, SNPA: 0012.da62.e0a8, State: Up, Holdtime: 7, Type: L1 Capable
  - Area Address(es): 00 11
  - IPv4 Address(es): 10.1.2.47*
  - IPv6 Address(es): fe80::212:dafe:fe62:e0a8*
  - Topologies: 'IPv4 Unicast', 'IPv6 Unicast'
  - Uptime: 01:09:33

Total neighbor count: 3

This table describes the significant fields shown in the display.

**Table 20: show isis instance isp neighbors Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System ID</td>
<td>Dynamic hostname of the system. The hostname is specified using the hostname command. If the dynamic hostname is not known or hostname dynamic disable command has been executed, the 6-octet system ID is used.</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface through which the neighbor is reachable.</td>
</tr>
<tr>
<td>SNPA</td>
<td>Data-link address (also known as the Subnetwork Point of Attachment [SNPA]) of the neighbor.</td>
</tr>
<tr>
<td>State</td>
<td>Adjacency state of the neighboring interface. Valid states are: Down, Init, and Up.</td>
</tr>
<tr>
<td>Holdtime</td>
<td>Hold time of the neighbor.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of adjacency.</td>
</tr>
<tr>
<td>IETF-NSF</td>
<td>Specifies whether the neighbor can adhere to the IETF-NSF restart mechanism. Valid states are Capable and Unable.</td>
</tr>
<tr>
<td>Area Address(es)</td>
<td>Number of area addresses on this router.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------</td>
</tr>
<tr>
<td>IPv4 Address(es)</td>
<td>IPv4 addresses configured on this router.</td>
</tr>
<tr>
<td>Topologies</td>
<td>Address and subaddress families for which IS-IS is configured.</td>
</tr>
<tr>
<td>Uptime</td>
<td>Time (in hh:mm:ss) that the neighbor has been up.</td>
</tr>
<tr>
<td>IPFRR: LFA Neighbor</td>
<td>IP fast reroute (IPFRR) loop-free alternate (LFA) neighbor.</td>
</tr>
<tr>
<td>LFA IPv4 address:</td>
<td>Address of the LFA.</td>
</tr>
<tr>
<td>LFA Interface:</td>
<td>LFA interface.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show isis neighbors` command with the `summary` keyword specified:

```
RP/0/RP0/CPU0:router# show isis instance isp neighbors summary
ISIS isp neighbor summary:
  State  L1  L2  L1L2
  Up     0   0   2
  Init   0   0   0
  Failed 0   0   0
```

This table describes the significant fields shown in the display.

### Table 21: `show isis neighbors summary` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>State</td>
<td>State of the neighbor is up, initialized, or failed.</td>
</tr>
<tr>
<td>L1</td>
<td>Number of Level 1 neighbors.</td>
</tr>
<tr>
<td>L2</td>
<td>Number of Level 2 neighbors.</td>
</tr>
<tr>
<td>L1L2</td>
<td>Number of Level 1 and 2 neighbors.</td>
</tr>
</tbody>
</table>
# show isis protocol

To display summary information about an Intermediate System-to-Intermediate System (IS-IS) instance, use the `show isis protocol` command in XR EXEC mode.

```
show isis [instance instance-id] protocol
```

## Syntax Description

- **instance instance-id** (Optional) Displays the IS-IS adjacencies for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.

## Command Default

No instance ID specified displays IS-IS adjacencies for all the IS-IS instances.

## Command Modes

XR EXEC mode

## Command History

- **Release**
  - Release 6.0  This command was introduced.

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read</td>
</tr>
</tbody>
</table>

## Examples

The following is sample output from the `show isis protocol` command:

```
RP/0/RP0/CPU0:router# show isis protocol

IS-IS Router: isp
    System Id: 0001.0000.0011
    IS Levels: level-1-2
    Manual area address(es):
        49

    Routing for area address(es):
        49
    Non-stop forwarding: Cisco Proprietary NSF Restart enabled
    Process startup mode: Cold Restart
    Topologies supported by IS-IS:
        IPv4 Unicast
            Level-1 ISPF status: Dormant (awaiting initial convergence)
            Level-2 ISPF status: Dormant (awaiting initial convergence)
            No protocols redistributed
            Distance: 115
    Interfaces supported by IS-IS:
        Loopback0 is running passively (passive in configuration)
        GigabitEthernet 0/4/0/1 is running actively (active in configuration)
        GigabitEthernet 0/5/0/1 is running actively (active in configuration)
```
This table describes the significant fields shown in the display.

Table 22: show isis protocol Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System ID:</td>
<td>Dynamic hostname of the system. The hostname is specified using the <code>hostname</code> command. If the dynamic hostname is not known or <code>hostname dynamic disable</code> command has been executed, the 6-octet system ID is used.</td>
</tr>
<tr>
<td>IS Levels:</td>
<td>IS-IS level of the router.</td>
</tr>
<tr>
<td>Manual area address(es)</td>
<td>Area addresses that are manually configured.</td>
</tr>
<tr>
<td>Routing for areaaddress(es)</td>
<td>Area addresses for which this router provides the routing.</td>
</tr>
<tr>
<td>Non-stop forwarding:</td>
<td>Status and name of nonstop forwarding (NSF).</td>
</tr>
<tr>
<td>Process startup mode:</td>
<td>Mode in which the last process startup occurred. Valid modes are:\n\n  • Cisco Proprietary NSF Restart\n  • IETF NSF Restart\n  • Cold Restart</td>
</tr>
<tr>
<td>iSPF status:</td>
<td>State of incremental shortest path first (iSPF) configuration for this IS-IS instance. Four states exist:\n\n  Disabled if iSPF has not been configured but is awaiting a full SPF to compile the topology for use by the iSPF algorithm.\n  Dormant if iSPF has been configured but is awaiting initial convergence before initializing.\n  Awake if iSPF has been configured but is awaiting a full SPF to compile the topology for use by the iSPF algorithm.\n  Active if IS-IS is ready to consider using the iSPF algorithm whenever a new route calculation needs to be run.</td>
</tr>
<tr>
<td>No protocols redistributed:</td>
<td>No redistributed protocol information exists to be displayed.</td>
</tr>
<tr>
<td>Distance:</td>
<td>Administrative distance for this protocol.</td>
</tr>
</tbody>
</table>
show isis route

To display IP reachability information for an Intermediate System-to-Intermediate System (IS-IS) instance, use the `show isis route` command in XR EXEC mode.

```
show isis [instance instance-id] [ {ipv4|ipv6|afi-all} ] [ {unicast | topology {all|topo-name}} | safi-all ] ]
route [ {ip-address mask|ip-address/length [longer-prefixes]} | summary ] [ backup ] [ detail ] [ sr-only ]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>instance</td>
<td>(Optional) Displays the IP reachability information for the specified IS-IS instance only.</td>
</tr>
<tr>
<td>instance-id</td>
<td>The <code>instance-id</code> argument is the instance identifier (alphanumeric) defined by the <code>router isis</code> command.</td>
</tr>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies IP Version 4 address prefixes.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td>afi-all</td>
<td>(Optional) Specifies all address prefixes.</td>
</tr>
<tr>
<td>unicast</td>
<td>(Optional) Specifies unicast address prefixes.</td>
</tr>
<tr>
<td>topology</td>
<td>(Optional) Specifies IS-IS paths to intermediate systems.</td>
</tr>
<tr>
<td>all</td>
<td>(Optional) Specifies all topologies.</td>
</tr>
<tr>
<td>topo-name</td>
<td>(Optional) Specifies topology table information and name of the topology table.</td>
</tr>
<tr>
<td>safi-all</td>
<td>(Optional) Specifies all secondary address prefixes.</td>
</tr>
<tr>
<td>ip-address</td>
<td>(Optional) Network IP address about which routing information should be displayed.</td>
</tr>
<tr>
<td>mask</td>
<td>(Optional) Network mask specified in either of two ways:</td>
</tr>
<tr>
<td>/ length</td>
<td>(Optional) Length of the IP prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash must precede the decimal value. Range is 0 to 32.</td>
</tr>
<tr>
<td>longer-prefixes</td>
<td>(Optional) Displays route and more-specific routes.</td>
</tr>
<tr>
<td>summary</td>
<td>(Optional) Displays topology summary information.</td>
</tr>
<tr>
<td>systemid</td>
<td>(Optional) Displays multicast information by system ID.</td>
</tr>
<tr>
<td>backup</td>
<td>(Optional) Displays backup information for this entry.</td>
</tr>
<tr>
<td>Command Default</td>
<td>Command Modes</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------------</td>
</tr>
<tr>
<td>No instance ID specified displays the IP reachability information for all the IS-IS instances.</td>
<td>XR EXEC mode</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show isis route` command:

```
RP/0/RSP0RP0/CPU0:router# show isis route

IS-IS lsp IPv4 Unicast routes
Codes: l1 - level 1, l2 - level 2, ia - interarea (leaked into level 1)
df - level 1 default (closest attached router), su - summary null
C - connected, S - static, R - RIP, B - BGP, O - OSPF
i - IS-IS (redistributed from another instance)

Maximum parallel path count: 8

L2 10.76.240.6/32 [4/115]
via 10.76.245.252, SRP0/1/0/2, isp2
via 10.76.246.252, SRP0/1/0/0, isp2
C 10.76.240.7/32
is directly connected, Loopback0
L2 10.76.240.9/32 [256/115]
via 10.76.249.2, tenGigE 0/3/0/0, isp3
L2 10.76.240.10/32 [296/115]
via 10.76.249.2, tenGigE 0/3/0/0, isp3
C 10.76.245.0/24
is directly connected, SRP0/1/0/2
C 10.76.246.0/24
is directly connected, SRP0/1/0/0
C 10.76.249.0/26
is directly connected, tenGigE 0/3/0/0
L2 10.101.10.0/24 [296/115]
via 10.76.249.2, tenGigE 0/3/0/0, isp3
```
This table describes the significant fields shown in the display.

**Table 23: show isis route ipv4 unicast Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C 172.18.0.0/24</td>
<td>Connected route for tenGigE interface 0/5/0/0.</td>
</tr>
<tr>
<td>C 172.19.0.0/24</td>
<td>Connected route for tenGigE interface 0/4/0/1.</td>
</tr>
<tr>
<td>L1 172.35.0.0/24 [10]</td>
<td>Level 1 route to network 172.35.0.0/24.</td>
</tr>
<tr>
<td>C 172.18.0/24</td>
<td>Connected route for loopback interface 0.</td>
</tr>
</tbody>
</table>

This is sample output from the *show isis route* command with **detail** keyword that shows prefix segment ID (SID) and Segment Routing Global Block (SRGB) values:

Sun May  4 13:05:11.073 PDT

L2 172.16.255.2/32 [10/115] medium priority  
  via 172.16.2.2, tenGigE 0/0/0/1, pe2 tag 255, SRGB Base: 16000, Weight: 0  
  src pe2.00-00, 172.16.255.2, tag 255, prefix-SID index 42, R:0 N:0 P:0  
  L1 adv [10] native, propagated, interarea, tag 255, prefix-SID index 42, R:0  
    N:0 P:0

This is sample output from the *show isis route* command with **sr-only** keyword that shows only routes associated with a segment routing prefix SID:

RP/0/RP0/CPU0:router# show isis route sr-only  

IS-IS 1 IPv4 Unicast routes  

Codes: L1 - level 1, L2 - level 2, ia - interarea (leaked into level 1)  
  df - level 1 default (closest attached router), su - summary null  
  C - connected, S - static, R - RIP, B - BGP, O - OSPF  
  A - access/subscriber, M - mobile, a - application  
  i - IS-IS (redistributed from another instance)

Maximum parallel path count: 8

C 20.1.0.100/32  
  is directly connected, Loopback0  
L2 20.1.0.101/32 [10/115]  
  via 10.1.1.101, GigabitEthernet0/0/0/2, r101, SRGB Base: 16000, Weight: 0  
L2 20.1.0.102/32 [30/115]  
  via 10.1.1.101, GigabitEthernet0/0/0/2, r101, SRGB Base: 16000, Weight: 0  
L2 20.1.0.103/32 [20/115]  
  via 10.4.1.103, GigabitEthernet0/0/0/1, r103, SRGB Base: 16000, Weight: 0
show isis spf-log

To display how often and why the router has run a full shortest path first (SPF) calculation, use the **show isis spf-log** command in XR EXEC mode.

```
show isis [instance instance-id] [[ipv4|ipv6|afi-all] [unicast [topology all|topo-name]|safi-all]]
spf-log [level {1|2}] [{ispf|fspf|prc|nhc}] [{detail|verbose}] [{last number|first number}]
```

**Syntax Description**

- `instance instance-id` *(Optional)*: Displays the IS-IS SPF log for the specified IS-IS instance only.
- `ipv4` *(Optional)*: Specifies IP Version 4 address prefixes.
- `ipv6` *(Optional)*: Specifies IP Version 6 address prefixes.
- `afi-all` *(Optional)*: Specifies all address prefixes.
- `unicast` *(Optional)*: Specifies unicast address prefixes.
- `multicast` *(Optional)*: Specifies multicast address prefixes.
- `topology all | topo-name` *(Optional)*: Specifies topology table information for all topologies or for the specified topology table (`top-name`).
- `safi-all` *(Optional)*: Specifies all secondary address prefixes.
- `level {1 | 2}` *(Optional)*: Displays the IS-IS SPF log for Level 1 or Level 2 independently.
- `ispf` *(Optional)*: Specifies incremental SPF entries only.
- `fspf` *(Optional)*: Specifies full SPF entries only.
- `prc` *(Optional)*: Specifies partial route calculations only.
- `nhc` *(Optional)*: Specifies next-hop route calculations only.
- `detail` *(Optional)*: Specifies detailed output. Includes a breakdown of the time taken to perform the calculation and changes resulting from the calculation.
- `verbose` *(Optional)*: Specifies verbose output.
- `last number` *(Optional)*: Specifies that the output is restricted to the last `number` of entries. Range is 1 to 210.
- `first number` *(Optional)*: Specifies that the output is restricted to the first `number` of entries. Range is 1 to 210.

**Command Default**

No instance ID specified displays IS-IS adjacencies for all the IS-IS instances.

Both Level 1 and Level 2 are configured if no level is specified.

Displays all types of route calculation (not just fspf, ispf and prc).

**Command Modes**

XR EXEC mode
**Command History**

**Release** | **Modification**
---|---
Release 6.0 | This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Examples**

The following is sample output from the `show isis spf-log` command:

```
RP/0/RP0/CPU0:router# show isis spf-log

IS-IS 1 Level 1 IPv4 Unicast Route Calculation Log

<table>
<thead>
<tr>
<th>Time</th>
<th>Total Trig</th>
<th>Node Count</th>
<th>First Trigger</th>
<th>LSP Triggers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Thurs Aug 19 2004</td>
<td></td>
<td>3</td>
<td>ensoft-grs7.00-00</td>
<td>LSPHEADER TLV CODE</td>
</tr>
<tr>
<td>12:00:50.787</td>
<td>FSPF</td>
<td>1</td>
<td>1</td>
<td>ensoft-grs7.00-00</td>
</tr>
<tr>
<td>12:00:52.846</td>
<td>FSPF</td>
<td>1</td>
<td>1</td>
<td>ensoft-grs7.00-00</td>
</tr>
<tr>
<td>12:00:56.049</td>
<td>FSPF</td>
<td>1</td>
<td>1</td>
<td>ensoft-grs7.00-00</td>
</tr>
</tbody>
</table>

IS-IS 1 Level 1 IPv4 Unicast Route Calculation Log

<table>
<thead>
<tr>
<th>Time</th>
<th>Total Trig</th>
<th>Node Count</th>
<th>First Trigger</th>
<th>LSP Triggers</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mon Aug 19 2004</td>
<td></td>
<td>4</td>
<td>ensoft-grs7.00-00</td>
<td>LSPHEADER TLV CODE</td>
</tr>
<tr>
<td>12:00:50.790</td>
<td>FSPF</td>
<td>1</td>
<td>1</td>
<td>ensoft-grs7.00-00</td>
</tr>
<tr>
<td>12:00:54.043</td>
<td>FSPF</td>
<td>1</td>
<td>1</td>
<td>ensoft-grs7.00-00</td>
</tr>
<tr>
<td>12:00:55.922</td>
<td>FSPF</td>
<td>1</td>
<td>2</td>
<td>ensoft-grs7.00-00</td>
</tr>
<tr>
<td>12:00:56.724</td>
<td>FSPF</td>
<td>1</td>
<td>13</td>
<td>ensoft-grs7.00-00</td>
</tr>
</tbody>
</table>
```

This table describes the significant fields shown in the display.

**Table 24: show isis spf-log ipv4 unicast Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>IS-IS level of the router.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>Time when the SPF calculation started.</td>
</tr>
<tr>
<td>Duration</td>
<td>Number of milliseconds taken to complete this SPF run. Elapsed time is wall clock time, not CPU time.</td>
</tr>
<tr>
<td>Nodes</td>
<td>Number of routers and pseudonodes (LANs) that make up the topology calculated in this SPF run.</td>
</tr>
</tbody>
</table>
Number of events that triggered this SPF run. When there is a topology change, often multiple link-state packets (LSPs) are received in a short time. Depending on the configuration of the `spf-interval` command, a router may wait for a fixed period of time before running a router calculation. This count denotes the number of triggering events that occurred while the router was waiting to run the calculation. For a full description of the triggering events, see List of Triggers.

LSP ID stored by the router whenever a full SPF calculation is triggered by the arrival of a new LSP. The LSP ID can suggest the source of routing instability in an area. If multiple LSPs are causing an SPF run, only the LSP ID of the first received LSP is remembered.

List of all reasons that triggered a full SPF calculation. For a list of possible triggers, see List of Triggers.

<table>
<thead>
<tr>
<th>Trigger</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PERIODIC</td>
<td>Runs a full SPF calculation very 15 minutes.</td>
</tr>
<tr>
<td>NEWLEVEL</td>
<td>Configured new level (using is-type) on this router.</td>
</tr>
<tr>
<td>RTCLEARED</td>
<td>Cleared IS-IS topology on the router.</td>
</tr>
<tr>
<td>MAXPATHCHANGE</td>
<td>Changed IP maximum parallel path.</td>
</tr>
<tr>
<td>NEWMETRIC</td>
<td>Changed link metric.</td>
</tr>
<tr>
<td>ATTACHFLAG</td>
<td>Changed Level 2 Attach bit.</td>
</tr>
<tr>
<td>ADMINDIST</td>
<td>Configured another administrative distance for the IS-IS instance on this router.</td>
</tr>
<tr>
<td>NEWADJ</td>
<td>Created a new adjacency to another router.</td>
</tr>
<tr>
<td>DELADJ</td>
<td>Deleted adjacency.</td>
</tr>
<tr>
<td>BACKUP</td>
<td>Installed backup route.</td>
</tr>
<tr>
<td>SEEDISPF</td>
<td>Seed incremental SPF.</td>
</tr>
<tr>
<td>NEXTHOP</td>
<td>Changed IP next-hop address.</td>
</tr>
<tr>
<td>NEWLSP0</td>
<td>New LSP 0 appeared in the topology.</td>
</tr>
<tr>
<td>LSPEXPIRED</td>
<td>Some LSP in the link-state database (LSDB) has expired.</td>
</tr>
<tr>
<td>LSPHEADER</td>
<td>Changed important LSP header fields.</td>
</tr>
<tr>
<td>TLVCODE</td>
<td>Type, length, and value (TLV) objects code mismatch, indicating that different TLV objects are included in the newest version of an LSP.</td>
</tr>
</tbody>
</table>
The following is sample output from the `show isis spf-log` command with the `first` keyword specified:

```
RP/0/RP0/CPU0:router# show isis spf-log first 2

IISIS isp Level 1 IPv4 Unicast Route Calculation Log
Time Total Trig
Timestamp Type (ms) Nodes Count First Trigger LSP Triggers
Mon Aug 16 2004
19:25:35.140 FSPF 1 1 1 12a5.00-00 NEWLSP0
19:25:35.646 FSPF 1 1 1 NEWADJ

IISIS isp Level 2 IPv4 Unicast Route Calculation Log
Time Total Trig
Timestamp Type (ms) Nodes Count First Trigger LSP Triggers
Mon Aug 16 2004
19:25:35.140 FSPF 1 1 1 12a5.00-00 NEWLSP0
19:25:35.347 FSPF 1 1 2 12a5.00-00 NEWSADJ TLVCODE
```

This table describes the significant fields shown in the display.

### Table 26: show isis spf-log first Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>IS-IS level of the router.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>Time at which the SPF calculation started.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of route calculation. The possible types are incremental SPF (iSPF),</td>
</tr>
<tr>
<td></td>
<td>full SPF (FSPF), or partial route calculation (PRC).</td>
</tr>
<tr>
<td>Time (ms)</td>
<td>Number of milliseconds taken to complete this SPF run. Elapsed time is</td>
</tr>
<tr>
<td></td>
<td>wall clock time, not CPU time.</td>
</tr>
<tr>
<td>Nodes</td>
<td>Number of routers and pseudonodes (LANs) that make up the topology</td>
</tr>
<tr>
<td></td>
<td>calculated in this SPF run.</td>
</tr>
</tbody>
</table>
Number of events that triggered this SPF run. When there is a topology change, often multiple link-state packets (LSPs) are received in a short time. Depending on the configuration of the `spf-interval` command, a router may wait for a fixed period of time before running a router calculation. This count denotes the number of triggering events that occurred while the router was waiting to run the calculation. For a full description of the triggering events, see *List of Triggers*.

LSP ID stored by the router whenever a full SPF calculation is triggered by the arrival of a new LSP. The LSP ID can suggest the source of routing instability in an area. If multiple LSPs are causing an SPF run, only the LSP ID of the first received LSP is remembered.

List of all reasons that triggered a full SPF calculation. For a list of possible triggers, see *List of Triggers*.

The following is sample output from the `show isis spf-log` command with the `detail` keyword specified:

```
RP/0/RP0/CPU0:router# show isis spf-log detail

IISIS isp Level 1 IPv4 Unicast Route Calculation Log
Time Total Trig
Timestamp Type (ms) Nodes Count First Trigger LSP Triggers
Mon Aug 16 2004 19:25:35.140 FSPF 1 1 1 12a5.00-00 NEWLSP0
Delay: 51ms (since first trigger)
SPT Calculation
CPU Time: 0ms
Real Time: 0ms
Prefix Updates
CPU Time: 1ms
Real Time: 1ms
New LSP Arrivals: 0
Next Wait Interval: 200ms

Results
Nodes: 1 0 1
Prefixes (Items)
Critical Priority: 0 0 0
High Priority: 0 0 0
Medium Priority 0 0 0
Low Priority 0 0 0
All Priorities 0 0 0
Prefixes (Routes)
Critical Priority: 0 - 0
High Priority: 0 - 0
Medium Priority 0 - 0
Low Priority: 0 - 0
All Priorities 0 - 0
```
This table describes the significant fields shown in the display.

**Table 27: show isis spf-log detail Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level</td>
<td>IS-IS level of the router.</td>
</tr>
<tr>
<td>Timestamp</td>
<td>Time at which the SPF calculation started.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of route calculation. The possible types are incremental SPF (iSPF), full SPF (FSPF), or partial route calculation (PRC).</td>
</tr>
<tr>
<td>Time (ms)</td>
<td>Number of milliseconds taken to complete this SPF run. Elapsed time is wall clock time, not CPU time.</td>
</tr>
<tr>
<td>Nodes</td>
<td>Number of routers and pseudonodes (LANs) that make up the topology calculated in this SPF run.</td>
</tr>
<tr>
<td>Trig Count</td>
<td>Number of events that triggered this SPF run. When there is a topology change, often multiple link-state packets (LSPs) are received in a short time. Depending on the configuration of the <code>spf-interval</code> command, a router may wait for a fixed period of time before running a router calculation. This count denotes the number of triggering events that occurred while the router was waiting to run the calculation. For a full description of the triggering events, see List of Triggers.</td>
</tr>
<tr>
<td>First Trigger LSP</td>
<td>LSP ID stored by the router whenever a full SPF calculation is triggered by the arrival of a new LSP. The LSP ID can suggest the source of routing instability in an area. If multiple LSPs are causing an SPF run, only the LSP ID of the first received LSP is remembered.</td>
</tr>
<tr>
<td>Triggers</td>
<td>List of all reasons that triggered a full SPF calculation. For a list of possible triggers, see List of Triggers.</td>
</tr>
<tr>
<td>Delay</td>
<td>Two different delays exist: 1. The delay between the time when the route calculation was first triggered and the time when it was run. 2. The delay between the end of the last route calculation and the start of this one. This is used to verify that the SPF-interval timers are working correctly, and is only reported for calculations after the first delay.</td>
</tr>
<tr>
<td>CPU Time</td>
<td>Two different CPU times exist: 1. CPU time (in milliseconds) taken to calculate the shortest path tree (SPT). 2. CPU time (in milliseconds) taken to perform the prefix updates.</td>
</tr>
<tr>
<td>Real Time</td>
<td>Two different real times exist: 1. Real time (in milliseconds) taken to calculate the shortest path tree (SPT). 2. Real time (in milliseconds) taken to perform the prefix updates.</td>
</tr>
<tr>
<td>New LSP Arrivals</td>
<td>Number of LSP arrivals since the start of this route calculation.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Next Wait Interval</td>
<td>Enforced delay until the next route calculation can be run, based on the <code>spf-interval</code> command configuration.</td>
</tr>
<tr>
<td>Reach</td>
<td>Number of reachable nodes or prefixes.</td>
</tr>
<tr>
<td>Unreach</td>
<td>Number of unreachable nodes or prefixes.</td>
</tr>
<tr>
<td>Total</td>
<td>Total number of nodes or prefixes at various priorities.</td>
</tr>
</tbody>
</table>
show isis statistics

To display Intermediate System-to-Intermediate System (IS-IS) traffic counters, use the `show isis statistics` command in XR EXEC mode.

```plaintext
show isis [instance instance-id] statistics [type interface-path-id]
```

**Syntax Description**

- **instance instance-id** (Optional) Displays the IS-IS traffic statistics for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.
- **type** Interface type. For more information, use the question mark (?) online help function.
- **interface-path-id** Physical interface or virtual interface.

**Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

**Command Default**

No instance ID specified displays IS-IS traffic statistics for all the IS-IS instances.

IS-IS traffic statistics are displayed for all interfaces.

**Command Modes**

XR EXEC mode

**Command History**

Release 6.0 This command was introduced.

**Usage Guidelines**

The `show isis statistics` command displays IS-IS traffic counters for the specified interface or all traffic counters if no interface is specified.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show isis statistics` command that shows all traffic counters:

```
RP/0/RP0/CPU0:router#show isis statistics
IS-IS isp statistics:
  Fast PSNP cache (hits/tries): 164115/301454
  Fast CSNP cache (hits/tries): 41828/43302
  Fast CSNP cache updates: 2750
  LSP checksum errors received: 0
  LSP Dropped: 1441
  SNP Dropped: 1958
```
UPD Max Queue size: 2431

Average transmit times and rate:
- **Hello**: 0 s, 987947 ns, 4/s
- **CSNP**: 0 s, 1452987 ns, 0/s
- **PSNP**: 0 s, 1531690 ns, 0/s
- **LSP**: 0 s, 1530018 ns, 1/s

Average process times and rate:
- **Hello**: 0 s, 874584 ns, 41/s
- **CSNP**: 0 s, 917925 ns, 29/s
- **PSNP**: 0 s, 1405458 ns, 0/s
- **LSP**: 0 s, 4352850 ns, 0/s

**Level-1:**
- LSPs sourced (new/refresh): 3376/2754
- Level-1:: LSPs sourced (new/refresh): SPF calculations: 3376/2754520 ISPF calculations
- IPv4 Unicast: OSPF calculations
- Next Hop Calculations: 5200 ISPF calculations
- Partial Route Calculations: 0

**IPv6 Unicast**
- SPF calculations: 527
- ISPF calculations: 0
- Next Hop Calculations: 13
- Partial Route Calculations: 1

**Level-2:**
- LSPs sourced (new/refresh): 4255/3332
- IPv4 Unicast
  - SPF calculations: 432
  - ISPF calculations: 0
  - Next Hop Calculations: 8
  - Partial Route Calculations: 4255/33320 LSPs sourced (new/refresh) SPF calculations: 4255/33320 ISPF calculations
- IPv6 Unicast
- SPF calculations: 4344
- ISPF calculations: 0
- Next-hop Next Hop Calculations: 1382
- Partial Route Calculations: 0

**Interface GigabitEthernet0/1/0/1.1:**
- Level-1 Hellos (sent/rcvd): 22398/25633
- Level-1 DR Elections: 66
- Level-1 LSPs (sent/rcvd): 246/7077
- Level-1 CSNPs (sent/rcvd): 0/3269
- Level-1 PSNPs (sent/rcvd): 22/0
- Level-1 LSP Flooding Duplicates: 25129
- Level-2 Hellos (sent/rcvd): 22393/67043
- Level-2 DR Elections: 55
- Level-2 LSPs (sent/rcvd): 265/437
- Level-2 CSNPs (sent/rcvd): 0/86750
- Level-2 PSNPs (sent/rcvd): 0/0
- Level-2 LSP Flooding Duplicates: 78690

This table describes the significant fields shown in the display.

**Table 28: show isis statistics Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fast PSNP cache (hits/tries)</td>
<td>Number of successful lookups (hits) along with the number of lookup attempts (tries). To save time or processing power when receiving multiple copies of the same LSP, IS-IS attempts to look up incoming LSPs to see if they have been received recently.</td>
</tr>
</tbody>
</table>
### Fast CSNP cache (hits/tries):
Number of successful lookups (hits) along with the number of lookup attempts (tries). To reduce CSNP construction time, IS-IS maintains a cache of CSNPs and attempts to look up CSNP in this cache before transmission on the interface.

### Fast CSNP cache updates:
Number of times the CSNP cache has been updated since the last clearing of statistics. The cache is updated on LSP addition or removal from the database.

### LSP checksum errors received:
Number of internal checksum errors received in LSPs.

### IIH (LSP/SNP) dropped:
Number of hello, LSP, and SNP messages dropped.

### IIH (UPD) Max Queue size:
Maximum number of queued packets.

### Average transmit times and rate:
Average time taken to transmit the pdu type across all interfaces and the corresponding rate at which the pdu type is being transmitted.

### Average process times and rate:
Average time taken to process an incoming pdu type across all interfaces and the corresponding rate at which the pdu type is being received.

### LSPs sourced (new/refresh):
Number of LSPs this IS-IS instance has created or refreshed. To find more details on these LSPs, use the `show isis lsp-log` command.

### SPF calculations:
Number of shortest path first (SPF) calculations. SPF calculations are performed only when the topology changes. They are not performed when external routes change. The interval at which SPF calculations are performed is configured using the `spf-interval` command.

### iSPF calculations:
Number of incremental shortest path first (iSPF) calculations. iSPF calculations are performed only when ISPF has been configured in the isis address family configuration submode.

### Partial Route Calculations:
Number of partial route calculations (PRCs). PRCs are processor intensive. Therefore, it may be useful to limit their number, especially how often a PRC is done, especially on slower networking devices. Increasing the PRC interval reduces the processor load on the router, but might slow the rate of convergence. The interval at which PRC calculations are performed is configured using the `spf-interval` command.

### Level-(1/2) (LSPs/CSNPs/PSNPs/Hellos) (sent/rcvd):
Number of LSPs, Complete Sequence Number Packets (CSNPs), Partial Sequence Number Packets (PSNPs), and hello packets sent or received on this interface.

### PTP Hellos (sent/rcvd):
Point-to-point (PTP) hellos sent and received.

### LSP Retransmissions:
Total number of retransmissions on each IS-IS LSP on a point-to-point interface. The LSP retransmission interval can be configured using the `retransmit-throttle-interval` command.

### Level-(1.2) DRElections:
Total number of Designated Intermediate System elections that have taken place. These counts are maintained on an individual level basis.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSP Flooding Duplicates:</td>
<td>Number of duplicate LSPs filtered from flooding to the neighbor. In case of parallel interfaces to the same neighbor, IS-IS optimizes the flooding by avoiding sending the same LSP copy on other interfaces.</td>
</tr>
</tbody>
</table>
show isis topology

To display a list of connected Intermediate System-to-Intermediate System (IS-IS) routers in all areas, use the `show isis topology` command in XR EXEC mode.

```
show isis [instance instance-id] [(ipv4|ipv6|afi-all)] [(unicast|topology [all|topo-name]|safi-all)][summary|level {1|2} [systemid system-id] [detail]
```

### Syntax Description

- **instance instance-id**: (Optional) Displays the IS-IS topology for the specified IS-IS instance only.
  - The `instance-id` argument is the instance identifier (alphanumeric) defined by the `router isis` command.
- **ipv4**: (Optional) Specifies IP Version 4 address prefixes.
- **ipv6**: (Optional) Specifies IP Version 6 address prefixes.
- **afi-all**: (Optional) Specifies all address prefixes.
- **unicast**: (Optional) Specifies unicast address prefixes.
- **topology topo-name**: (Optional) Specifies topology table information and name of the topology table.
- **safi-all**: (Optional) Specifies all secondary address prefixes.
- **summary**: (Optional) Displays a brief list of the IS-IS topology.
- **level { 1 | 2 }**: (Optional) Displays the IS-IS link-state topology for Level 1 or Level 2 independently.
- **systemid system-id**: (Optional) Displays the information for the specified router only.
- **detail**: (Optional) Displays detailed information on the IS-IS topology.

### Command Default

No instance ID specified displays a list of connected routers in all areas for all the IS-IS instances. Both Level 1 and Level 2 is configured if no level is specified.

### Command Modes

- XR EXEC mode

### Command History

- **Release 6.0** This command was introduced.

### Usage Guidelines

Use the `show isis topology` command to verify the presence and connectivity among all routers in all areas.

### Task ID

- **Task ID**
  - **isis read**
Examples

The following is sample output from the `show isis topology` command:

```
RP/0/RP0/CPU0:router# show isis topology

IS-IS isp paths to (Level-1) routers
System Id  Metric  Next-hop Interface  SNPA
ensoft-5   10      ensoft-5      PO0/4/0/1  *PtoP*
ensoft-5   10      ensoft-5      G10/5/0/0  0003.6cff.0680
ensoft-11  --      --            --

IS-IS isp paths to (Level-2) routers
System Id  Metric  Next-hop Interface  SNPA
ensoft-5   10      ensoft-5      PO0/4/0/1  *PtoP*
ensoft-5   10      ensoft-5      G10/5/0/0  0003.6cff.0680
ensoft-11  --      --            --
```

This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System ID</td>
<td>Dynamic hostname of the system. The hostname is specified using the <code>hostname</code> command. If the dynamic hostname is not known or <code>hostname dynamic disable</code> command has been executed, the 6-octet system ID is used.</td>
</tr>
<tr>
<td>Metric</td>
<td>Metric assigned to the link and used to calculate the cost from each router using the links in the network to other destinations. Range is 1 to 16777214. Default is 1 to 63 for narrow metric and 1 to 16777214 for wide metric. 0 is set internally if no metric has been specified by the user.</td>
</tr>
<tr>
<td>Next-hop</td>
<td>Address of the next-hop.</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface used to reach the neighbor.</td>
</tr>
<tr>
<td>SNPA</td>
<td>Data-link address (also known as the Subnetwork Point of Attachment [SNPA]) of the neighbor.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show isis topology` command with the `summary` keyword specified:

```
RP/0/RP0/CPU0:router# show isis topology summary

IS-IS 10 IS Topology Summary IPv4 Unicast

+-------------+-----------+-----------+-------------+-----------+-----------
|              | Reach     | UnReach   | Total       | Reach     | UnReach   |
| L1           |           |           |             |           |           |
| Router nodes | 1         | 1         | 2           | 1         | 1         |
| Pseudo nodes | 0         | 0         | 0           | 0         | 0         |
| Total nodes  | 1         | 1         | 2           | 1         | 1         |
```
This table describes the significant fields shown in the display.

**Table 30: show isis topology summary Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>L1/L2</td>
<td>IS-IS level of the router.</td>
</tr>
<tr>
<td>Reach</td>
<td>Number of router nodes or pseudonodes that are reachable.</td>
</tr>
<tr>
<td>UnReach</td>
<td>Number of router nodes or pseudonodes that are unreachable.</td>
</tr>
<tr>
<td>Total</td>
<td>Total number of reachable and unreachable nodes.</td>
</tr>
</tbody>
</table>
show protocols (IS-IS)

To group a number of protocol show commands according to the specified address family, use the `show protocols` command in XR EXEC mode.

```
show protocols [{afi-all|ipv4|ipv6}] [{all|protocol}]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>afi-all</td>
<td>(Optional) Specifies all address families.</td>
</tr>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies an IPv4 address family.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies an IPv6 address family.</td>
</tr>
<tr>
<td>all</td>
<td>(Optional) Specifies all protocols for a given address family.</td>
</tr>
<tr>
<td>protocol</td>
<td>(Optional) Specifies a routing protocol. For the IPv4 address family, the options are:</td>
</tr>
<tr>
<td></td>
<td>• bgp</td>
</tr>
<tr>
<td></td>
<td>• isis</td>
</tr>
<tr>
<td></td>
<td>• ospf</td>
</tr>
<tr>
<td></td>
<td>• rip</td>
</tr>
<tr>
<td></td>
<td>For the IPv6 address family, the options are:</td>
</tr>
<tr>
<td></td>
<td>• bgp</td>
</tr>
<tr>
<td></td>
<td>• isis</td>
</tr>
<tr>
<td></td>
<td>• ospfv3</td>
</tr>
</tbody>
</table>

**Command Default**

If no address family is specified, the default is IPv4.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If IPv6 is enabled on an IS-IS instance, the instance is displayed in the `show protocols ipv6` command output. IPv4 IS-IS instances are displayed in the `show protocols ipv4` command output.

When using the `show protocols` command with the `ipv6` or `ipv4` keyword, you get all routing instances in that particular address family—not only IS-IS instances.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read</td>
</tr>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>
Examples

The following example shows the output for the `show protocols` command:

```
RP/0/RP0/CPU0:router# show protocols ipv4

IS-IS Router: uut
System Id: 0000.0000.12a8
IS Levels: level-1-2
Manual area address(es):
  49.1515.1515
Routing for area address(es):
  49.1515.1515
Non-stop forwarding: Disabled
Most recent startup mode: Cold Restart
Topologies supported by IS-IS:
  IPv4 Unicast
    Level-1
      Metric style (generate/accept): Narrow/Narrow
      ISPF status: Disabled
    Level-2
      Metric style (generate/accept): Narrow/Narrow
      ISPF status: Disabled
    Redistributing:
      static
      Distance: 115
  IPv6 Unicast
    Level-1
      ISPF status: Disabled
    Level-2
      ISPF status: Disabled
    No protocols redistributed
    Distance: 45
Interfaces supported by IS-IS:
  GigabitEthernet 0/6/0/0 is running actively (active in configuration)
```

This table describes the significant fields shown in the display.

**Table 31: show protocols ipv4 Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System ID</td>
<td>Dynamic hostname of the system. The hostname is specified using the <code>hostname</code> command. If the dynamic hostname is not known or <code>hostname dynamic disable</code> command has been executed, the 6-octet system ID is used.</td>
</tr>
<tr>
<td>IS Levels</td>
<td>IS-IS level of the router.</td>
</tr>
<tr>
<td>Manual area address(es)</td>
<td>Area addresses configured manually on the originating router.</td>
</tr>
<tr>
<td>Routing for area address(es)</td>
<td>Area addresses for which this router provides the routing.</td>
</tr>
<tr>
<td>Non-stop forwarding</td>
<td>Status and name of NSF.</td>
</tr>
<tr>
<td>Most recent startup mode</td>
<td>Mode in which the most recent startup was performed.</td>
</tr>
<tr>
<td>Topologies supported by IS-IS</td>
<td>Address and subaddress family IS-IS are configured.</td>
</tr>
</tbody>
</table>
### Field Description

**Metric style**
Type, length, and value (TLV) objects accepted by IS-IS. To configure this value, see the `metric-style narrow`, on page 52, `metric-style transition`, on page 53, or `metric-style wide`, on page 54 command.

**ISPF status**
State of ISPF configuration for this IS-IS instance. Four states exist:
- **Disabled** if ISPF has not been configured but is awaiting a full SPF to compile the topology for use by the ISPF algorithm.
- **Dormant** if ISPF has been configured but is awaiting initial convergence before initializing.
- **Awake** if ISPF has been configured but is awaiting a full SPF to compile the topology for use by the ISPF algorithm.
- **Active** if IS-IS is ready to consider using the ISPF algorithm whenever a new route calculation needs to be run.

**Redistributing**
IS-IS is configured to redistribute IP static routes into Level 1 or Level 2. The `redistribute` command is used to configure redistribution.

**Distance**
Administrative distance.

**Interfaces supported by IS-IS**
Interfaces and their states currently supported by IS-IS. Both operational and configuration status are displayed.

The following example shows how to disable the IPv4 address family, with no output shown for IS-IS IPv4 instances from the `show protocols ipv4` command:

```plaintext
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router isis uut
RP/0/RP0/CPU0:router(config-isis)# no address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis)# commit

RP/0/RP0/CPU0:router# show protocols ipv4
```
shutdown (IS-IS)

To disable the Intermediate System-to-Intermediate System (IS-IS) protocol on a particular interface, use the `shutdown` command in interface configuration mode. To re-enable the IS-IS protocol, use the `no` form of this command.

```
shutdown
no shutdown
```

**Command Default**

IS-IS protocol is enabled.

**Command Modes**

Interface configuration

**Command History**

Release 6.0   This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Examples**

The following example disables the IS-IS protocol on tenGigE interface 0/1/0/1:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-isis-if)# shutdown
```
**single-topology**

To configure the link topology for IP Version 4 (IPv4) when IP Version 6 (IPv6) is configured, use the `single-topology` command in address family configuration mode. To remove the `single-topology` command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
single-topology
no single-topology
```

**Command Default**

Performs in multitopology mode in which independent topologies for IPv4 and IPv6 are running in a single area or domain.

**Command Modes**

IPv6 address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `single-topology` command to allow Intermediate System-to-Intermediate System (IS-IS) for IPv6 to be configured on interfaces along with an IPv4 network protocol. All interfaces must be configured with the identical set of network protocols, and all routers in the IS-IS area (for Level 1 routing) or the domain (for Level 2 routing) must support the identical set of network layer protocols on all interfaces.

When single-topology support for IPv6 is being used, only old-style type, length, and value (TLV) objects may be used and a single shortest path (SPF) individual level is used to compute IPv4 (if configured) and IPv6 routes. The use of a single SPF means that both IPv4 IS-IS and IPv6 IS-IS routing protocols must share a network topology.

To allow link information to be shared between IPv4 and IPv6, you must configure the `single-topology` command for an address family. In single-topology IPv6 mode, the configured metric is always the same for both IPv4 and IPv6.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to enable single-topology mode for IPv6:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# net 49.0000.0000.0001.00
RP/0/RP0/CPU0:router(config-isis)# address-family ipv6 unicast
RP/0/RP0/CPU0:router(config-isis-af)# single-topology
```
**snmp-server traps isis**

To enable the Simple Network Management Protocol (SNMP) server notifications (traps) available for IS-IS, use the `snmp-server traps isis` command in XR Config mode. To disable all available SNMP notifications, use the `no` form of this command.

```
snmp-server traps isis {all | traps set}
no snmp-server traps isis {all | traps set}
```

**Syntax Description**

- `all` Specifies all IS-IS SNMP server traps.
- `traps set` Specify any set of trap names.

**Command Default**

SNMP server traps notification is disabled.

**Command Modes**

XR Config mode

**Command History**

- **Release** 6.0  This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- **Task** ID read, write
- **Operations**

**Examples**

The following examples show how to enable all SNMP server traps available for isis:

```
RP/0/RP0/CPU0:router(config)# snmp-server traps isis?
adjacency-change isisAdjacencyChange
all isisAll
area-mismatch isisAreaMismatch
attempt-to-exceed-max-sequence isisAttemptToExceedMaxSequence
authentication-failure isisAuthenticationFailure
authentication-type-failure isisAuthenticationTypeFailure
corrupted-lsp-detected isisCorruptedLSPDetected
database-overload isisDatabaseOverload
database-error isisDatabaseError
database-full isisDatabaseFull
database-inconsistent isisDatabaseInconsistent
database-open isisDatabaseOpen
database-read-only isisDatabaseReadOnly
database-read-write isisDatabaseReadWrite
database-write-only isisDatabaseWriteOnly
database-write-write isisDatabaseWriteWrite
database-update isisDatabaseUpdate
database-update-failed isisDatabaseUpdateFailed
database-update-function isisDatabaseUpdateFunction
database-update-in progress isisDatabaseUpdateInProgress
database-update-done isisDatabaseUpdateDone
database-update-done-failed isisDatabaseUpdateDoneFailed
database-update-done-function isisDatabaseUpdateDoneFunction
database-update-done-in-progress isisDatabaseUpdateDoneInProgress
database-update-done-in-progress-failed isisDatabaseUpdateDoneInProgressFailed
database-update-done-in-progress-function isisDatabaseUpdateDoneInProgressFunction
database-update-done-in-progress-function-failed isisDatabaseUpdateDoneInProgressFunctionFailed
database-update-done-in-progress-function-in-progress isisDatabaseUpdateDoneInProgressFunctionInProgress
database-update-done-in-progress-function-in-progress-failed isisDatabaseUpdateDoneInProgressFunctionInProgressFailed
database-update-done-in-progress-function-in-progress-function isisDatabaseUpdateDoneInProgressFunctionInProgressFunction
database-update-done-in-progress-function-in-progress-function-in-progress isisDatabaseUpdateDoneInProgressFunctionInProgressFunctionInProgress
```
The following examples show how to enable area-mismatch lsp-error-detected trap:

RP/0/RP0/CPU0:router(config)# snmp-server traps isis area-mismatch lsp-error-detected
**spf-interval**

To customize IS-IS throttling of shortest path first (SPF) calculations, use the `spf-interval` command in address family configuration mode. To restore default values, use the `no` form of this command.

```
spf-interval [{initial-wait initial}] [secondary-wait secondary] [maximum-wait maximum] ... [level {1|2}]
no spf-interval [{initial-wait initial}] [secondary-wait secondary] [maximum-wait maximum] ... [level {1|2}]
```

**Syntax Description**

- **initial-wait initial**: Initial SPF calculation delay (in milliseconds) after a topology change. Range is 0 to 120000.
- **secondary-wait secondary**: Hold time between the first and second SPF calculations (in milliseconds). Range is 0 to 120000.
- **maximum-wait maximum**: Maximum interval (in milliseconds) between two consecutive SPF calculations. Range is 0 to 120000.
- **level {1|2}**: (Optional) Enables the SPF interval configuration for Level 1 or Level 2 independently.

**Command Default**

- `initial-wait initial`: 50 milliseconds
- `secondary-wait secondary`: 200 milliseconds
- `maximum-wait maximum`: 5000 milliseconds

**Command Modes**

Address family configuration

**Command History**

Release 6.0  This command was introduced.

**Usage Guidelines**

SPF calculations are performed only when the topology changes. They are not performed when external routes change.

Use the `spf-interval` command to control how often the software can perform the SPF calculation. The SPF calculation is processor intensive. Therefore, it may be useful to limit how often this calculation is done, especially when the area is large and the topology changes often. Increasing the SPF interval reduces the processor load of the router, but potentially slows the rate of convergence.

**Task ID**

```
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>
```
The following example shows how to set the initial SPF calculation delay to 10 milliseconds and the maximum interval between two consecutive SPF calculations to 5000 milliseconds:

RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# spf-interval initial-wait 10 maximum-wait 5000
**spf prefix-priority (IS-IS)**

To assign a priority to an ISIS prefix for customizing the RIB update sequence, use the `spf prefix-priority` command in address family configuration mode. To restore default values, use the `no` form of this command.

```
spf prefix-priority [level {1|2}] {critical|high|medium} {access-list-name|tag tag}
no spf prefix-priority [level {1|2}] {critical|high|medium} [{access-list-name|tag tag}]
```

**Syntax Description**

- `level {1 | 2}` (Optional) Enables the assignment of a priority to Level 1 or Level 2 independently.
- `critical` Assigns a critical priority.
- `high` Assigns a high priority.
- `medium` Assigns a medium priority.
- `access-list-name` Name of an access list.
- `tag tag` Specifies a tag to indicate priority. The `tag` argument range is 1 to 4294967295.

**Command Default**

By default, IPv4 prefixes with a length of 32 and IPv6 prefixes with a length of 128 are given medium priority. The remaining prefixes are given low priority.

**Command Modes**

Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `spf prefix-priority` command to change the sequence of prefix updates to the RIB after an SPF is run. ISIS installs prefixes in the RIB according to the following priority order:

- Critical > High > Medium > Low

The `spf prefix-priority` command supports prefix lists for the first three priorities. The unmatched prefixes are updated with low priority.

If a `spf prefix-priority` is specified, the default behavior of prioritizing either length 32 or 128 prefixes for IPv4 or IPv6, respectively, as `medium` is disabled.

**Examples**

The following example shows how to set the prefix priorities:

```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list isis-critical-acl
RP/0/RP0/CPU0:router(config-ipv4_pfx)# 10 permit 0.0.0.0 eq 32
```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list isis-med-acl
RP/0/RP0/CPU0:router(config-ipv4_pfx)# 10 permit 0.0.0.0/0 eq 29

RP/0/RP0/CPU0:router(config)# ipv4 prefix-list isis-high-acl
RP/0/RP0/CPU0:router(config-ipv4_pfx)# 10 permit 0.0.0.0/0 eq 30

RP/0/RP0/CPU0:router(config)# router isis ring
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-af)# spf prefix-priority critical isis-critical-acl
RP/0/RP0/CPU0:router(config-isis-af)# spf prefix-priority high isis-high-acl
RP/0/RP0/CPU0:router(config-isis-af)# spf prefix-priority medium isis-med-acl
summary-prefix (IS-IS)

To create aggregate addresses for the Intermediate System-to-Intermediate System (IS-IS) protocol, use the `summary-prefix` command in address family configuration mode. To restore the default behavior, use the `no` form of this command.

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>address</td>
<td>Summary address designated for a range of IPv4 addresses. The <code>address</code> argument must be in four-part, dotted-decimal notation.</td>
</tr>
<tr>
<td>/ prefix-length</td>
<td>Length of the IPv4 or IPv6 prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash must precede the decimal value.</td>
</tr>
<tr>
<td>ipv6-prefix</td>
<td>Summary prefix designated for a range of IPv6 prefixes. The <code>ipv6-prefix</code> argument must be in the form documented in RFC 2373, in which the address is specified in hexadecimal using 16-bit values between colons.</td>
</tr>
<tr>
<td>level { 1</td>
<td>2 }</td>
</tr>
<tr>
<td>tag tag</td>
<td>Sets a tag value. The value range is 1-4294967295.</td>
</tr>
</tbody>
</table>

**Command Default**

All redistributed routes are advertised individually.

Both Level 1 and Level 2 are configured if no level is specified.

**Command Modes**

Address family configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Multiple groups of addresses can be summarized for a given level. Routes learned from other routing protocols can also be summarized. The metric used to advertise the summary is the smallest metric of all the more-specific routes. Use the `summary-prefix` command to help reduce the size of the routing table.

This command also reduces the size of the link-state packets (LSPs) and thus the link-state database. It also helps ensure stability, because a summary advertisement depends on many more specific routes. If one more-specific route flaps, in most cases, this flap does not cause a flap of the summary advertisement.

The drawback of summary addresses is that other routes might have less information to calculate the most optimal routing table for all individual destinations.

**Note**

When IS-IS advertises a summary prefix, it automatically inserts the summary prefix into the IP routing table but labels it as a “discard” route entry. Any packet that matches the entry is discarded to prevent routing loops. When IS-IS stops advertising the summary prefix, the routing table entry is removed.
### Examples

The following example shows how to redistribute Open Shortest Path First (OSPF) routes into IS-IS:

In the OSPF routing table, IPv6 routes exist for 3ffe:f000:0001:0000::/64, 3ffe:f000:0002:0000::/64, 3ffe:f000:0003:0000::/64, and so on. This example shows only 3ffe:f000::/24 advertised into IPv6 IS-IS Level 2.

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# address-family ipv4 ipv6 unicast
RP/0/RP0/CPU0:router(config-isis-af)# redistribute ospf ospfv3 2 level-2
RP/0/RP0/CPU0:router(config-isis-af)# summary-prefix 10.10.10.10 3ffe:f000::/24 level-2
RP/0/RP0/CPU0:router(config-isis-af)# summary-prefix 10.10.10.10 3ffe:f000::/24 tag
```
To allow an IS-IS interface to participate in forming adjacencies without advertising connected prefixes in the system link-state packets (LSPs), use the `suppressed` command in interface configuration mode. To enable advertising connected prefixes, use the `no` form of this command.

```
suppressed
no suppressed
```

**Command Default**
Interface is active.

**Command Modes**
Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `suppressed` command to reduce the number of routes that IS-IS has to maintain, improving convergence times after an isolated failure. Improvement is noticeable if the command is used widely throughout the network. Other routers in the domain do not install routes to the affected connected prefixes.

**Examples**
The following example shows how to disable the advertisement of connected prefixes on tenGigE interface 0/1/0/1:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-isis-if)# suppressed
```
tag (IS-IS)

To associate and advertise a tag with the prefix of an IS-IS interface, use the `tag` command in interface address family configuration mode. To restore the default behavior, use the `no` form of this command.

```
tag [tag]
no tag [tag]
```

**Syntax Description**
- `tag` Interface tag. Range is 1 to 4294967295.

**Command Default**
Default is that no tag is associated and advertised.

**Command Modes**
Interface address family configuration

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**
- **Task ID**
  - **Operations**
    - `isis` read, write

**Examples**
The following example shows how to associate and advertise an interface tag:

```
RP/0/RP0/CPU0:router(config)# router isis isp
RP/0/RP0/CPU0:router(config-isis)# interface tenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-isis-if)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-isis-if-af)# tag 234
```
**topology-id**

To differentiate one topology in the domain from another while configuring a multicast routing table, use the `topology-id` command in Intermediate System-to-Intermediate System (IS-IS) address family configuration submode. To disable the topology use the `no` form of the command.

```
topology-id  isis-multicast-topology-id-number
no  topology-id  isis-multicast-topology-id-number
```

**Syntax Description**
- `isis-multicast-topology-id-number`: ID number for a specific IS-IS multicast topology. Range is 6 to 4095.

**Command Default**
No topology is associated with a routing table by default.

**Command Modes**
IS-IS address family configuration

**Command History**
- **Release 6.0** This command was introduced.

**Task ID**
```
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>
```

**Examples**
The following example shows how to differentiate a topology from another in the multicast routing table in IS-IS routing:

```
RP/0/RP0/CPU0:router(config)#  router  isis  isp
RP/0/RP0/CPU0:router(config-isisp)#  address-family  ipv4  multicast  topology  green
RP/0/RP0/CPU0:router(config-isisp-af)#  topology-id  2666
```
trace (IS-IS)

To set the IS-IS buffer size, use the `trace` command in XR Config mode. To return to the default value, use the `no` form of this command.

```
trace [{detailed|severe|standard}] max-trace-entries
no trace [{detailed|severe|standard}]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>detailed</code></td>
<td>Specifies the buffer size for detailed traces. Range is detailed</td>
</tr>
<tr>
<td><code>severe</code></td>
<td>Specifies the buffer size for severe traces. Range is severe</td>
</tr>
<tr>
<td><code>standard</code></td>
<td>Specifies the buffer size for standard traces. Range is standard</td>
</tr>
<tr>
<td><code>max-trace-entries</code></td>
<td>Sets the maximum number of trace entries. Range is 1-20000</td>
</tr>
</tbody>
</table>

**Command Default**

None

**Command Modes**

Router IS-IS configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the isis buffer size for severe traces to 1200:

```
RP/0/RP0/CPU0#router isis isp
RP/0/RP0/CPU0#router(config-isis)#trace severe 1200
```
trace (IS-IS)
OSPF Commands

This module describes the commands used to configure and monitor the Open Shortest Path First (OSPF) routing protocol.

For detailed information about OSPF concepts, configuration tasks, and examples, see the Implementing OSPF on Cisco NCS 5000 Series Routers module in the *Routing Configuration Guide for Cisco NCS 5000 Series Routers*.

Note

Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

- address-family (OSPF), on page 156
- adjacency stagger, on page 157
- area (OSPF), on page 159
- authentication (OSPF), on page 160
- authentication-key (OSPF), on page 162
- auto-cost (OSPF), on page 164
- capability opaque disable, on page 165
- clear ospf process, on page 166
- clear ospf redistribution, on page 167
- clear ospf routes, on page 168
- clear ospf statistics, on page 169
- cost (OSPF), on page 170
- cost-fallback (OSPF), on page 172
- database-filter all out (OSPF), on page 174
- dead-interval (OSPF), on page 175
- default-cost (OSPF), on page 177
- default-information originate (OSPF), on page 178
- default-metric (OSPF), on page 180
- demand-circuit (OSPF), on page 181
- disable-dn-bit-check, on page 182
- distance (OSPF), on page 183
- distance ospf, on page 185
- distribute-list, on page 187
• domain-id (OSPF), on page 189
• domain-tag, on page 191
• flood-reduction (OSPF), on page 192
• hello-interval (OSPF), on page 193
• ignore lsa mospf, on page 195
• interface (OSPF), on page 196
• log adjacency changes (OSPF), on page 198
• loopback stub-network , on page 199
• max-lsa, on page 200
• max-metric, on page 202
• maximum interfaces (OSPF), on page 205
• maximum paths (OSPF), on page 206
• maximum redistributed-prefixes (OSPF), on page 207
• message-digest-key, on page 208
• mpls ldp auto-config (OSPF), on page 211
• mpls ldp sync (OSPF), on page 212
• mtu-ignore (OSPF), on page 213
• multi-area-interface, on page 215
• neighbor (OSPF), on page 217
• neighbor database-filter all out, on page 219
• network (OSPF), on page 220
• nsf (OSPF), on page 222
• nsf flush-delay-time (OSPF), on page 224
• nsf interval (OSPF), on page 225
• nsf lifetime (OSPF), on page 226
• nssa (OSPF), on page 227
• ospf name-lookup, on page 229
• packet-size (OSPF), on page 230
• passive (OSPF), on page 232
• priority (OSPF), on page 234
• protocol shutdown, on page 236
• queue dispatch flush-lsa, on page 237
• queue dispatch incoming, on page 238
• queue dispatch rate-limited-lsa, on page 239
• queue dispatch spf-lsa-limit, on page 240
• queue limit, on page 241
• range (OSPF), on page 242
• redistribute (OSPF), on page 244
• retransmit-interval (OSPF), on page 248
• route-policy (OSPF), on page 250
• router-id (OSPF), on page 251
• router ospf, on page 253
• sham-link, on page 254
• show ospf, on page 255
• show ospf border-routers, on page 257
• show ospf database, on page 259
• show ospf flood-list, on page 272
• show ospf interface, on page 274
• show ospf message-queue, on page 276
• show ospf neighbor, on page 278
• show ospf request-list, on page 285
• show ospf retransmission-list, on page 287
• show ospf routes, on page 289
• show ospf sham-links, on page 294
• show ospf summary-prefix, on page 296
• show ospf virtual-links, on page 298
• show protocols (OSPF), on page 300
• snmp context (OSPF), on page 302
• snmp trap (OSPF), on page 304
• snmp trap rate-limit (OSPF), on page 305
• spf prefix-priority (OSPFv2), on page 306
• stub (OSPF), on page 308
• summary-prefix (OSPF), on page 309
• timers lsa group-pacing, on page 311
• timers lsa min-arrival, on page 312
• timers throttle lsa all (OSPF), on page 313
• timers throttle spf (OSPF), on page 315
• transmit-delay (OSPF), on page 317
• virtual-link (OSPF), on page 319
• vrf (OSPF), on page 321
address-family (OSPF)

To enter address family configuration mode for Open Shortest Path First (OSPF), use the `address-family` command in the appropriate mode. To disable address family configuration mode, use the `no` form of this command.

```
address-family ipv4 [unicast]
no address-family ipv4 [unicast]
```

**Syntax Description**
- `ipv4` Specifies IP Version 4 (IPv4) address prefixes.
- `unicast` (Optional) Specifies unicast address prefixes.

**Command Default**
An address family is not specified.

**Command Modes**
Router configuration.

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
OSPF version 2 automatically provides routing services for IPv4 unicast topologies, so this command is redundant.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to configure the OSPF router process with IPv4 unicast address prefixes:

```
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# address-family ipv4 unicast
```
adjacency stagger

To configure staggering of OSPF adjacency during reload, process restart, and process clear, use the adjacency stagger command in XR Config mode. To turn off adjacency staggering, either use the disable keyword or use the no form of this command.

```
adjacency stagger {disable|initial-num-nbr max-num-nbr}
no adjacency stagger
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disable</td>
<td>Disables adjacency staggering.</td>
</tr>
<tr>
<td>initial-num-nbr</td>
<td>The initial number of simultaneous neighbors allowed to form adjacency to FULL in any area to bring up to FULL after a router reload, OSPF process restart, or OSPF process clear. Range is 1-65535. Default is 2.</td>
</tr>
<tr>
<td>max-num-nbr</td>
<td>The subsequent number of simultaneous neighbors allowed to form adjacency, per OSPF instance, after the initial set of OSPF neighbors have become FULL. Range is 1-65535. Default is 64.</td>
</tr>
</tbody>
</table>

**Command Default**

OSPF adjacency staggering is enabled.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Staggering of the OSPF adjacency during reload, process restart (without NSR or graceful-restart), and process clear reduces the overall adjacency convergence time.

Initially, allow 2 (configurable) neighbors to form adjacency to FULL per area. After the first adjacency reaches FULL, up to 64 (configurable) neighbors can form adjacency simultaneously for the OSPF instance (all areas). However, areas without any FULL adjacency is restricted by the initial area limit.

**Note**

Adjacency stagger and OSPF nonstop forwarding (NSF) are mutually exclusive. Adjacency stagger will not be activated if `nsf` is configured under router ospf configuration.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure adjacency stagger for a 2 neighbors initially and for a maximum of 3 neighbors:
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# adjacency stagger 2 3
area (OSPF)

To configure an Open Shortest Path First (OSPF) area, use the **area** command in the appropriate mode. To terminate an OSPF area, use the **no** form of this command.

```
area  area-id
no  area  area-id
```

**Syntax Description**

- **area-id**: Identifier of an OSPF area. The *area-id* argument can be specified as either a decimal value or an IP address (dotted decimal) format. Range is 0 to 4294967295.

**Command Default**

No OSPF area is defined.

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **area** command to explicitly configure an area. Commands configured under the area configuration mode (such as the **interface** [OSPF] and **authentication** commands), are automatically bound to that area.

To modify or remove the area, the *area-id* argument format must be the same as the format used when creating the area. Otherwise, even if the actual 32-bit value matches, the area is not matched. For example, if you create an area with an *area-id* of 10 it would not match an *area-id* of 0.0.0.10.

**Note**

To remove the specified area from the router configuration, use the **no area  area-id** command. The **no area  area-id** command removes the area and all area options, such as **authentication**, **default-cost**, **nssa**, **range**, **stub**, **virtual-link**, and **interface**.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure area 0 and tenGigE interface 0/2/0/0. tenGigE interface 0/2/0/0 is bound to area 0 automatically.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/2/0/0
```
authentication (OSPF)

To enable plain text, Message Digest 5 (MD5) authentication, or null authentication for an Open Shortest Path First (OSPF) interface, use the authentication command in the appropriate mode. To remove such authentication, use the no form of this command.

```
authentication [{message-digest [keychain keychain]null}]  
no authentication
```

**Syntax Description**
- `message-digest` (Optional) Specifies that MD5 is used.
- `keychain keychain` (Optional) Specifies a keychain name.
- `null` (Optional) Specifies that no authentication is used. Useful for overriding password or MD5 authentication if configured for an area.

**Command Default**
If this command is not specified in interface configuration mode, then the interface adopts the authentication parameter specified by the area.
If this command is not specified in area configuration mode, then the interface adopts the authentication parameter specified for the process.
If this command is not specified at any level, then the interface does not use authentication.
If no keyword is specified, plain text authentication is used.

**Command Modes**
- Interface configuration
- Area configuration
- Router configuration
- Virtual-link configuration
- Multi-area interface configuration
- Sham-link configuration

**Command History**
- Release 6.0 This command was introduced.

**Usage Guidelines**
Use the authentication command to specify an authentication type for the interface, which overrides the authentication specified for the area to which this interface belongs. If this command is not included in the configuration file, the authentication configured in the area to which the interface belongs is assumed (as specified by the area authentication command).

The authentication type and password must be the same for all OSPF interfaces that are to communicate with each other through OSPF. If you specified plain text authentication, use the authentication-key command to specify the plain text password.

If you enable MD5 authentication with the message-digest keyword, you must configure a key with the message-digest-key interface command.
To manage the rollover of keys and enhance MD5 authentication for OSPF, you can configure a container of keys called a keychain with each key comprising the following attributes: generate/accept time, key identification, and authentication algorithm. The keychain management feature is always enabled.

Note

Changes to the system clock will impact the validity of the keys in the existing configuration.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

Examples

The following example shows how to set authentication for areas 0 and 1 of OSPF routing process 201. Authentication keys are also provided.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
RP/0/RP0/CPU0:router(config-ospf)# router-id 10.1.1.1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# authentication
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# authentication-key mykey
RP/0/RP0/CPU0:router(config-ospf-ar-if)# exit
RP/0/RP0/CPU0:router(config-ospf)# area 1
RP/0/RP0/CPU0:router(config-ospf-ar)# authentication
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/0
RP/0/RP0/CPU0:router(config-ospf-ar-if)# authentication-key mykey1
```

The following example shows how to configure use of an authentication keychain:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
RP/0/RP0/CPU0:router(config-ospf)# router-id 10.1.1.1
RP/0/RP0/CPU0:router(config-ospf)# authentication message-digest keychain mykeychain
```
authentication-key (OSPF)

To assign a password to be used by neighboring routers that are using the Open Shortest Path First (OSPF) simple password authentication, use the authentication-key command in the appropriate mode. To remove a previously assigned OSPF password, use the no form of this command.

```
authentication-key [clear|encrypted] password
no authentication-key
```

**Syntax Description**

- **clear** (Optional) Specifies that the key be clear text.
- **encrypted** (Optional) Specifies that the key be encrypted using a two-way algorithm.
- **password** Any contiguous string up to 8 characters in length that can be entered from the keyboard. For example, mypswd2.

**Command Default**

- If this command is not specified in interface configuration mode, then the interface adopts the OSPF password parameter specified by the area.
- If this command is not specified in area configuration mode, then the interface adopts the OSPF password parameter specified for the process.
- If this command is not specified at any level, then no password is specified.
- Clear is the default if the clear or encrypted keyword is not specified.

**Command Modes**

- Interface configuration
- Area configuration
- Router configuration
- Virtual-link configuration
- Multi-area configuration
- Sham-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The password created by this command is inserted directly into the OSPF header when the Cisco IOS XR software originates routing protocol packets. A separate password can be assigned to each network on an individual interface basis. All neighboring routers on the same network must have the same password to be able to exchange OSPF information.

The authentication-key command must be used with the authentication command. If the authentication command is not configured, the password provided by the authentication-key command is ignored and no authentication is adopted by the OSPF interface.
The **authentication-key** command cannot be used with the **authentication** command when the **message-digest** or **null** keyword is configured.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to configure an authentication password as the string `yourpass`:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
RP/0/RP0/CPU0:router(config-ospf)# authentication-key yourpass
```
auto-cost (OSPF)

To control how the Open Shortest Path First (OSPF) protocol calculates default metrics for the interface, use the `auto-cost` command in the appropriate mode. To revert to the default reference bandwidth, use the `no` form of this command.

```
auto-cost {reference-bandwidth mbps|disable}
no auto-cost {reference-bandwidth|disable}
```

**Syntax Description**
- `reference-bandwidth mbps`: Specifies a rate in Mbps (bandwidth). Range is 1 to 4294967.
- `disable`: Assigns a cost based on interface type.

**Command Default**
- `mbps`: 100 Mbps

**Command Modes**
- Router configuration

**Command History**
- **Release Modification**
  - Release 6.0  This command was introduced.

**Usage Guidelines**
By default OSPF calculates the OSPF metric for an interface according to the bandwidth of the interface. The OSPF metric is calculated as the `mbps` value divided by bandwidth, with `mbps` equal to 108 by default.
If you have multiple links with high bandwidth (such as OC-192), you might want to use a larger number to differentiate the cost on those links. That is, the metric calculated using the default `mbps` value is the same for all high-bandwidth links.

Recommended usage of cost configuration for OSPF interfaces with high bandwidth is to be consistent: Either explicitly configure (by using the `cost` command) or choose the default (by using the `auto-cost` command).

The value set by the `cost` command overrides the cost resulting from the `auto-cost` command.

**Examples**
The following example shows how to set the reference value for the auto cost calculation to 1000 Mbps:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# auto-cost reference-bandwidth 1000
```
capability opaque disable

To prevent Multiprotocol Label Switching traffic engineering (MPLS TE) topology information flooded to the network through opaque LSAs, use the `capability opaque disable` command in the appropriate mode. To restore MPLS TE topology information flooded through opaque LSAs to the network, use the `no` form of the command.

```plaintext
capability opaque disable
no capability opaque disable
```

**Command Default**
Opaque LSAs are allowed.

**Command Modes**
XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
The `capability opaque disable` command prevents flooded MPLS TE information (Types 1 and 4) through opaque LSAs of all scope (Types 9, 10, and 11).

Control opaque LSA support capability must be enabled for OSPF to support MPLS TE.

The MPLS TE topology information is flooded to the area through opaque LSAs by default.

**Task ID**
Task ID Operations

<table>
<thead>
<tr>
<th>ID</th>
<th>read, write</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td></td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to prevent OSPF from supporting opaque services:

```plaintext
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# capability opaque disable
```
clear ospf process

To reset an Open Shortest Path First (OSPF) router process without stopping and restarting it, use the `clear ospf process` command in XR EXEC mode.

```
clear ospf [process-name] process
```

**Syntax Description**

`process-name` (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only the specified routing process is affected. Otherwise, all OSPF processes are reset.

**Command Default**

No default behavior or value

**Command Modes**

XR EXEC mode

**Command History**

| Release 6.0 | This command was introduced. |

**Usage Guidelines**

When the OSPF router process is reset, OSPF releases all resources allocated, cleans up the internal database, and shuts down and restarts all interfaces that belong to the process.

**Note**

The `clear ospf process` command may change the router ID unless the OSPF router ID is explicitly configured through the `router-id (OSPF)`, on page 251 command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to reset all OSPF processes:

```
RP/0/RP0/CPU0:router# clear ospf process
```

The following example shows how to reset the OSPF 1 process:

```
RP/0/RP0/CPU0:router# clear ospf 1 process
```
clear ospf redistribution

To clear all routes redistributed from other protocols out of the Open Shortest Path First (OSPF) routing table, use the **clear ospf redistribution** command in XR EXEC mode.

```
clear ospf [process-name] redistribution
```

**Syntax Description**
- **process-name** (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only the specified routing process is affected. Otherwise, all OSPF routes are cleared.

**Command Default**
No default behavior or value

**Command Modes**
XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the **clear ospf redistribution** command to cause the routing table to be read again. OSPF regenerates and sends Type 5 and Type 7 link-state advertisements (LSAs) to its neighbors. If an unexpected route has appeared in the OSPF redistribution, using this command corrects the issue.

**Note**
Use of this command can cause a significant number of LSAs to flood the network. We recommend that you use this command with caution.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to clear all redistributed routes across all processes from other protocols:

```
RP/0/RP0/CPU0:router# clear ospf redistribution
```
clear ospf routes

To clear all Open Shortest Path First (OSPF) routes from the OSPF routing table, use the `clear ospf routes` command in XR EXEC mode.

```
clear ospf [process-name] routes
```

**Syntax Description**
- `process-name` (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only the specified routing process is affected. Otherwise, all OSPF routes are cleared.

**Command Default**
No default behavior or value

**Command Modes**
XR EXEC mode

**Command History**
- **Release** 6.0 This command was introduced.

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**
- **Task ID** Operations
  - ospf read, write

**Examples**
The following example shows how to clear all OSPF routes from the OSPF routing table and recalculate valid routes. When the OSPF routing table is cleared, OSPF routes in the global routing table are also recalculated.

```
RP/0/RP0/CPU0:router# clear ospf routes
```
clear ospf statistics

To clear the Open Shortest Path First (OSPF) statistics of neighbor state transitions, use the `clear ospf statistics` command in XR EXEC mode.

```
clear ospf [process-name ] statistics [neighbor [type interface-path-id] [ip-address]]
```

**Syntax Description**

- **process-name** (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only the specified routing process is affected. Otherwise, all OSPF statistics of neighbor state transitions are cleared.

- **neighbor** (Optional) Clears the state transition counters of the specified neighbor only.

- **type** (Optional) Interface type. For more information, use the question mark (?) online help function.

- **interface-path-id** (Optional) Physical interface or virtual interface.

  Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

  For more information about the syntax for the router, use the question mark (?) online help function.

- **ip-address** (Optional) IP address of a specified neighbor for whom you want to clear the state transition counter.

**Command Default**

No default behavior or value

**Command Modes**

XR EXEC mode

**Command History**

- **Release** Modification
  - Release 6.0 This command was introduced.

**Usage Guidelines**

Use the `clear ospf statistics` command to reset OSPF counters. Reset is useful to detect changes in counter values.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to reset the OSPF transition state counters for all neighbors on Packet-over-SONET/SDH (POS) interface 0/2/0/0:

```
RP/0/RP0/CPU0:router# clear ospf statistics neighbor POS 0/2/0/0
```
**cost (OSPF)**

To explicitly specify the interface (network) for Open Shortest Path First (OSPF) path calculation, use the `cost` command in the appropriate mode. To remove the cost, use the `no` form of this command.

```
cost  cost
no  cost
```

**Syntax Description**

`cost`  Unsigned integer value expressed as the link-state metric. Range is 1 to 65535.

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the cost parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the cost parameter specified for the process.

If this command is not specified at any level, then the cost is calculated by the `auto-cost` command.

**Command Modes**

- Interface configuration
- Area configuration
- Router configuration
- Multi-area configuration
- Sham-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The link-state metric is advertised as the link cost in the router link advertisement. Cisco IOS XR software does not support type of service (ToS), so you can assign only one cost for each interface.

In general, the path cost is calculated using the following formula:

\[ \text{cost} = \frac{108}{\text{bandwidth}} \]  

This calculation is the default reference bandwidth used by the auto-costing calculation which establishes the interface auto-cost. The `auto-cost` command can set this reference bandwidth to some other value. The `cost` command is used to override the auto-costing calculated default value for interfaces.

Using this formula, the default path cost is 1 for any interface that has a link bandwidth of 100 Mbps or higher. If this value does not suit the network, configure the reference bandwidth for auto calculating costs based on the link bandwidth.

The value set by the `cost` command overrides the cost resulting from the `auto-cost (OSPF)` command.
### Examples

The following example shows how to set the cost value to 65 for tenGigE interface 0/1/0/1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# cost 65
```
**cost-fallback (OSPF)**

To apply higher cost than the normal interface cost when the cumulative bandwidth of a bundle interface goes below the threshold specified and to revert to the original cost if the cumulative bandwidth goes above the configured threshold, use the `cost-fallback` command. To remove the cost-fallback, use the `no` form of this command.

```
cost-fallback cost threshold bandwidth
no cost-fallback
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>cost</code></td>
<td>Unsigned integer value expressed as the link-state metric. Range is 1 to 65535, but typically, cost-fallback value is supposed to be set to a value higher than the normal cost.</td>
</tr>
<tr>
<td><code>threshold</code></td>
<td>Unsigned integer value expressed in Mbits per second. Range is 1 to 4294967.</td>
</tr>
<tr>
<td><code>bandwidth</code></td>
<td></td>
</tr>
</tbody>
</table>

**Command Default**

If this command is not specified in interface configuration mode, the currently effective interface cost takes effect even when the cumulative bandwidth goes down below the maximum bandwidth. Unlike the interface cost command, this cost-fallback command is available only under interface configuration mode; it is not available in area or process level. Unlike other interface specific parameters, no inheritance will take place from area or process level if this command is not specified at interface level.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The fallback cost must be set to a higher value than the normal interface cost. The motivation of setting the fallback cost is to cost out an interface or disfavor an interface without shutting it down when its cumulative bandwidth goes below the user specified threshold, so that the traffic can take an alternative path. The normal interface cost will take over when the cumulative bandwidth reaches or exceeds user-specified threshold.

**Examples**

The following example shows how to set the cost-fallback value:

The following example shows how to set the cost-fallback value for Bundle-Ether:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 100
RP/0/RP0/CPU0:router(config-ospf)# router-id 2.2.2.2
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface bundle-Ether
RP/0/RP0/CPU0:router(config-ospf-ar-if)# cost-fallback 1000 threshold 300
```
cost-fallback (OSPF)
database-filter all out (OSPF)

To filter outgoing link-state advertisements (LSAs) to an Open Shortest Path First (OSPF) interface, use the `database-filter all out` command in the appropriate mode. To restore the forwarding of LSAs to the interface, use the `disable` form of the command.

`database-filter all out [disable|enable]`

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Command Default</th>
<th>Command Modes</th>
<th>Command History</th>
<th>Usage Guidelines</th>
<th>Examples</th>
</tr>
</thead>
</table>
| disable (Optional) Disables filtering.     | The database filter is disabled. | Interface configuration | Release 6.0  This command was introduced. | No specific guidelines impact the use of this command. | The following example shows how to prevent flooding of OSPF LSAs to broadcast, nonbroadcast, and point-to-point networks reachable through tenGigE interface 0/1/0/1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# database-filter all out
```
dead-interval (OSPF)

To set the interval after which a neighbor is declared dead when no hello packets are observed, use the `dead-interval` command in the appropriate mode. To return to the default time, use the `no` form of this command.

```
dead-interval seconds
no dead-interval
```

**Syntax Description**

| seconds | Integer that specifies the interval (in seconds). Range is 1 to 65535. The value must be the same for all nodes on the network. |

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the dead interval parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the dead interval parameter specified for the process.

If this command is not specified at any level, then the dead interval is four times the interval set by the `hello-interval` (OSPF) command.

**Command Modes**

- Interface configuration
- Area configuration
- Router configuration
- Virtual-link configuration
- Multi-area configuration
- Sham-link configuration

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

The dead interval value must be the same for all routers and access servers on a specific network.

If the hello interval is configured, the dead interval value must be larger than the hello interval value. The dead interval value is usually configured four times larger than the hello interval value.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the OSPF dead interval to 40 seconds:
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# dead-interval 40
default-cost (OSPF)

To specify a cost for the default summary route sent into a stub area or not-so-stubby area (NSSA), use the `default-cost` command in area configuration mode. To remove the assigned default route cost, use the `no` form of this command.

```
default-cost  cost
no  default-cost  cost
```

**Syntax Description**
- `cost`: Cost for the default summary route used for a stub or NSSA area. The acceptable value is a 24-bit number.

**Command Default**
- `cost`: 1

**Command Modes**
- Area configuration

**Command History**

```
Release 6.0  This command was introduced.
```

**Usage Guidelines**

Use the `default-cost` command only on an Area Border Router (ABR) attached to a stub or an NSSA area.

In all routers and access servers attached to the stub area, the area should be configured as a stub area using the `stub` command in the area submode. Use the `default-cost` command only on an ABR attached to the stub area. The `default-cost` command provides the metric for the summary default route generated by the ABR into the stub area.

**Examples**

The following example shows how to assign a default cost of 20 to a stub area. The tenGigE interface 0/4/0/3 is also configured in the stub area):

```
RP/0/RP0/CPU0:router#  configure
RP/0/RP0/CPU0:router(config)#  router ospf 201
RP/0/RP0/CPU0:router(config-ospf)#  area 10.15.0.0
RP/0/RP0/CPU0:router(config-ospf-ar)#  stub
RP/0/RP0/CPU0:router(config-ospf-ar)#  default-cost 20
RP/0/RP0/CPU0:router(config-ospf-ar)#  interface tenGigE 0/4/0/3
```
default-information originate (OSPF)

To generate a default external route into an Open Shortest Path First (OSPF) routing domain, use the `default-information originate` command in the appropriate mode. To disable this feature, use the `no` form of this command.

```
default-information originate [always] [metric metric-value] [metric-type type-value] [route-policy policy-name] [tag tag-value]
no default-information originate
```

**Syntax Description**
- `always` (Optional) Always advertises the default route regardless of whether the routing table has a default route.
- `metric metric-value` (Optional) Specifies the metric used for generating the default route. The default metric value is 1. Range is 1 to 16777214.
- `metric-type type-value` (Optional) Specifies the external link type associated with the default route advertised into the OSPF routing domain. It can be one of the following values:
  1—Type 1 external route  
  2—Type 2 external route
- `tag tag-value` (Optional) 32-bit dotted-decimal value attached to each external route. This is not used by the OSPF protocol itself. It may be used to communicate information between autonomous system boundary routers (ASBRs). If a tag is not specified, then the configured OSPF process number is used.
- `route-policy policy-name` (Optional) Specifies that a routing policy be used and the routing policy name.

**Command Default**
When you do not use this command in router configuration mode, no default external route is generated into an OSPF routing domain.
- `metric-value`: 1  
- `type-value`: 2  
- `tag-value`: configured OSPF process number

**Command Modes**
Router configuration

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Whenever you use the `redistribute` or `default-information originate` command to redistribute routes into an OSPF routing domain, the software automatically becomes an Autonomous System Boundary Router (ASBR). However, an ASBR does not, by default, generate a default route into the OSPF routing domain. The software still must have a default route for itself before it generates one, except when you have specified the `always` keyword.
The `default-information originate` route-policy attach point conditionally injects the default route 0.0.0.0/0 into the OSPF link-state database, and is done by evaluating the attached policy. If any routes specified in the policy exist in the global RIB, then the default route is inserted into the link-state database. If there is no match condition specified in the policy, the policy passes and the default route is generated into the link-state database.

For information about the default-information originate attach point, see the *OSPF Policy Attach Points* section in the Implementing Routing Policy chapter in Routing Configuration Guide for Cisco NCS 5000 Series Routers.

For information about routing policies, see the Routing Policy Commands chapter in the Routing Command Reference for Cisco NCS 5000 Series Routers.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to specify a metric of 100 for the default route redistributed into the OSPF routing domain and an external metric type of Type 1:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospf 109
RP/0/RP0/CPU0:router(config)#redistribute igrp 108 metric 100
RP/0/RP0/CPU0:router(config-ospf)#default-information originate metric 100 metric-type 1
```
default-metric (OSPF)

To set default metric values for routes redistributed from another protocol into the Open Shortest Path First (OSPF) protocol, use the **default-metric** command in the appropriate mode. To return to the default state, use the **no** form of this command.

```
default-metric  value
no default-metric  value
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>value</td>
<td>Default metric value appropriate for the specified routing protocol. Range is 1 to 16777214.</td>
</tr>
</tbody>
</table>

**Command Default**

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

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **default-metric** command with the **redistribute** command to cause the current routing protocol to use the same metric value for all redistributed routes. A default metric helps solve the problem of redistributing routes with incompatible metrics. Whenever metrics do not convert, use a default metric to provide a reasonable substitute and enable the redistribution to proceed.

The default-metric value configured in OSPF configuration does not apply to connected routes that are redistributed to OSPF using the **redistribute connected** command. To set a non-default metric for connected routes, configure OSPF with the **redistribute connected metric metric-value** command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to advertise Intermediate System-to-Intermediate System (IS-IS) protocol-derived routes into OSPF and assign a metric of 10:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# default-metric 10
RP/0/RP0/CPU0:router(config-ospf)# redistribute isis IS-IS_isp
```
demand-circuit (OSPF)

To configure the Open Shortest Path First (OSPF) protocol to treat the interface as an OSPF demand circuit, use the demand-circuit command in the appropriate mode. To remove the demand circuit designation from the interface, use the no form of this command.

```
demand-circuit [{disable|enable}]
no demand-circuit
```

**Syntax Description**
- `disable` (Optional) Disables the interface as an OSPF demand circuit.
- `enable` (Optional) Enables the interface as an OSPF demand circuit.

**Command Default**
- If this command is not specified in interface configuration mode, then the interface adopts the demand circuit parameter specified by the area.
- If this command is not specified in area configuration mode, then the interface adopts the demand circuit parameter specified for the process.
- If this command is not specified at any level, then the circuit is not a demand circuit.

**Command Modes**
- Interface configuration
- Area configuration
- Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

On point-to-point interfaces, only one end of the demand circuit must be configured with this command. Periodic hello messages are suppressed and periodic refreshes of link-state advertisements (LSAs) do not flood the demand circuit. Use the demand-circuit command to allow the underlying data link layer to be closed when the topology is stable. In point-to-multipoint topology, only the multipoint end must be configured with this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the configuration for an OSPF demand circuit:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-router)# demand-circuit
```
disable-dn-bit-check

To specify that down bits should be ignored, use the `disable-dn-bit-check` command in VPN routing and forwarding (VRF) configuration mode. To specify that down bits should be considered, use the `no` form of this command.

```
disable-dn-bit-check
no disable-dn-bit-check
```

**Command Default**

Down bits are considered.

**Command Modes**

VRF configuration mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to specify that down bits be ignored:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# vrf v1
RP/0/RP0/CPU0:router(config-ospf-vrf)# disable-dn-bit-check
```
distance (OSPF)

To define an administrative distance, use the distance command in an appropriate configuration mode. To remove the distance command from the configuration file and restore the system to its default condition in which the software removes a distance definition, use the no form of this command.

```
distance weight [ip-address wildcard-mask [access-list-name]]
no distance weight ip-address wildcard-mask [access-list-name]
```

**Syntax Description**
- **weight**: Administrative distance. Range is 10 to 255. Used alone, the weight argument specifies a default administrative distance that the software uses when no other specification exists for a routing information source. Routes with a distance of 255 are not installed in the routing table. lists the default administrative distances. Table 32: Default Administrative Distances, on page 183
- **ip-address**: (Optional) IP address in four-part, dotted-decimal notation.
- **wildcard-mask**: (Optional) Wildcard mask in four-part, dotted decimal format. A bit set to 1 in the mask argument instructs the software to ignore the corresponding bit in the address value.
- **access-list-name**: (Optional) Name of an IP access list to be applied to incoming routing updates.

**Command Default**
If this command is not specified, then the administrative distance is the default, as specified in Table 32: Default Administrative Distances, on page 183.

**Command Modes**
Router configuration

**Command History**
- **Release 6.0**: This command was introduced.

**Usage Guidelines**
An administrative distance is an integer from 10 to 255. In general, the higher the value, the lower the trust rating. An administrative distance of 255 means that the routing information source cannot be trusted at all and should be ignored. Weight values are subjective; no quantitative method exists for choosing weight values.

If an access list is used with this command, it is applied when a network is being inserted into the routing table. This behavior allows you to filter networks based on the IP prefix supplying the routing information. For example, you could filter possibly incorrect routing information from networking devices not under your administrative control.

The order in which you enter distance commands can affect the assigned administrative distances in unexpected ways (see the “Examples” section for further clarification).

This table lists default administrative distances.

```
Table 32: Default Administrative Distances

<table>
<thead>
<tr>
<th>Route Source</th>
<th>Default Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Connected interface</td>
<td>0</td>
</tr>
</tbody>
</table>
```
### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

In the following example, the `router ospf` command sets up OSPF routing instance 1. The first `distance` command sets the default administrative distance to 255, which instructs the software to ignore all routing updates from networking devices for which an explicit distance has not been set. The second `distance` command sets the administrative distance for all networking devices on the Class C network 192.168.40.0 0.0.0.255 to 90.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# distance 255
RP/0/RP0/CPU0:router(config-ospf)# distance 90 192.168.40.0 0.0.0.255
```
distance ospf

To define Open Shortest Path First (OSPF) route administrative distances based on route type, use the **distance ospf** command in router configuration mode. To restore the default value, use the **no** form of this command.

**distance ospf** {intra-area|inter-area|external} **distance**

**no distance ospf**

**Syntax Description**

<table>
<thead>
<tr>
<th>intra-area</th>
<th>inter-area</th>
<th>external</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sets the type of area. It can be one of the following values:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>intra-area — All routes within an area.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>inter-area — All routes from one area to another area.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>external — All routes from other routing domains, learned by redistribution.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Any combination of the above areas is allowed.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Command Default**

**distance**: 110

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You must specify one of the keywords.

Use the **distance ospf** command to perform the same function as the **distance** command used with an access list. However, the **distance ospf** command sets a distance for an entire group of routes, rather than a specific route that passes an access list.

A common reason to use the **distance ospf** command is when you have multiple OSPF processes with mutual redistribution, and you want to prefer internal routes from one over external routes from the other.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to change the external distance to 200, making the route less reliable:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# redistribute ospf 2
RP/0/RP0/CPU0:router(config-ospf)# distance ospf external 200
RP/0/RP0/CPU0:router(config-ospf)# exit
```
RP/0/RP0/CPU0:router(config)# router ospf 2
RP/0/RP0/CPU0:router(config-ospf)# redistribute ospf 1
RP/0/RP0/CPU0:router(config-ospf)# distance ospf external 200
distribute-list

To filter networks received or transmitted in Open Shortest Path First (OSPF) updates, use the `distribute-list` command in the appropriate mode. To change or cancel the filter, use the `no` form of this command.

```
distribute-list  {access-list-name  {in|out  [bgp number|connected|ospf instance|static]}|route-policy route-policy-name  in}
no distribute-list  {access-list-name  {in|out}|route-policy route-policy-name  in}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>access-list-name</code></td>
<td>Standard IP access list name. The list defines which networks are to be received and which are to be suppressed in routing updates.</td>
</tr>
<tr>
<td><code>in</code></td>
<td>Applies the access list or route-policy to incoming routing updates.</td>
</tr>
<tr>
<td><code>out</code></td>
<td>Applies the access list to outgoing routing updates. The <code>out</code> keyword is available only in router configuration mode.</td>
</tr>
<tr>
<td><code>bgp</code></td>
<td>(Optional) Applies the access list to BGP routes.</td>
</tr>
<tr>
<td><code>connected</code></td>
<td>(Optional) Applies the access list to connected routes.</td>
</tr>
<tr>
<td><code>ospf</code></td>
<td>(Optional) Applies the access list to OSPF routes (not the current OSPF process).</td>
</tr>
<tr>
<td><code>static</code></td>
<td>(Optional) Applies the access list to statically configured routes.</td>
</tr>
<tr>
<td><code>route-policy</code></td>
<td>Specifies the route-policy to filter OSPF prefixes.</td>
</tr>
<tr>
<td><code>route-policy-name</code></td>
<td></td>
</tr>
</tbody>
</table>

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the distribute list parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the distribute list parameter specified for the process.

If this command is not specified at any level, then the distribute list is disabled.

**Command Modes**

- Interface configuration
- Area configuration
- Router configuration
- Multi-area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `distribute-list` command to limit which OSPF routes are installed on this router. The `distribute-list` command does not affect the OSPF protocol itself.
The **distribute-list in** is configurable at instance (process), area, and interface levels. Regular OSPF configuration inheritance applies. Configuration is inherited from instance > area > interface levels.

Use the **route-policy route-policy-name** keyword and argument to allow use of route policies to filter OSPF prefixes.

---

**Note**

Either an access-list, or a route-policy can be used in a single command, not both. Configuring the command with access-list removes the route-policy configuration, and vice versa.

The "if tag..." statements can be used in **distribute-list in route-policy**. The matching on route tag supports operators "eq/ge/is/le". Operator "in" is not supported.

---

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to prevent OSPF routes from the 172.17.10.0 network from being installed if they are learned in area 0:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ipv4 access-list 3
RP/0/RP0/CPU0:router(config-ipv4-acl)# deny 172.17.10.0 0.0.0.255
RP/0/RP0/CPU0:router(config-ipv4-acl)# permit any any
!
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# distribute-list 3 in
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/3
```
domain-id (OSPF)

To specify the Open Shortest Path First (OSPF) VPN routing and forwarding (VRF) domain ID, use the domain-id command in VRF configuration mode. To remove an OSPF VRF domain ID, use the no form of this command.

```
domain-id [secondary] type [{0005|0105|0205|8005}] value value
no domain-id [secondary] type [{0005|0105|0205|8005}] value value
```

**Syntax Description**

- **secondary** (Optional) OSPF secondary domain ID.
- **type** Primary OSPF domain ID in hex format.
- **value** OSPF domain ID value in hex format (six octets).

**Command Default**

No domain ID is specified.

**Command Modes**

VRF configuration mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

An OSPF domain ID must be explicitly configured. The OSPF domain ID helps OSPF determine how to translate a prefix received through Border Gateway Protocol (BGP) from the remote provider edge (PE). If the domain IDs match, OSPF generates a Type 3 link state advertisement (LSA). If the domain IDs do not match, OSPF generates a Type 5 LSA.

There is only one primary domain ID. There can be multiple secondary domain IDs.

**Note**

When an IOS XR router and an IOS router are configured as peers, the two Domain IDs must match. Manually configure the IOS XR Domain ID value to match the IOS default Domain ID value. This ensures that the routes have route code "OIA" because they are learned as inter-area routes. If the Domain IDs do not match, the routes have route code, "O-E2" because they are learned as external routes. Use the `show ip ospf` command to get the OSPF Domain ID from the IOS router. Then, set the IOS XR Domain ID to the same value using the `domain-id` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to specify a domain ID:
configure
router ospf 1
vrf v1
domain-id type 0105 value AABCCDDEEFF
domain-tag

To specify the Open Shortest Path First (OSPF) VPN routing and forwarding (VRF) domain tag, use the `domain-tag` command in VRF configuration mode. To remove an OSPF VRF domain tag, use the `no` form of this command.

```
domain-tag  tag
no domain-tag
```

**Syntax Description**

- **tag** OSPF domain tag as a 32-bit value. The valid range is 0 to 4294967295.

**Command Default**

No OSPF VRF domain tag is specified.

**Command Modes**

- VRF configuration mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The domain tag is added to any Type 5 link state advertisements (LSAs) generated as a result of VPN-IP routes received from Border Gateway Protocol (BGP). The domain-tag is derived from BGP autonomous system number (ASN).

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to specify the domain tag:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 01
RP/0/RP0/CPU0:router(config-ospf)# domain-tag 234
```
flood-reduction (OSPF)

To suppress the unnecessary flooding of link-state advertisements (LSAs) in stable topologies, use the flood-reduction command in the appropriate mode. To remove this functionality from the configuration, use the no form of this command.

```
flood-reduction [enable|disable]
no flood-reduction [enable|disable]
```

**Syntax Description**
- **enable** (Optional) Turns on this functionality at a specific level.
- **disable** (Optional) Turns off this functionality at a specific level.

**Command Default**
If this command is not specified in interface configuration mode, then the interface adopts the flood reduction parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the flood reduction parameter specified for the process.

If this command is not specified at any level, then flood reduction is disabled.

**Command Modes**
- Interface configuration
- Area configuration
- Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
All routers supporting the OSPF demand circuit are compatible and can interact with routers supporting flooding reduction.

**Examples**
The following example shows how to reduce the flooding of unnecessary LSAs for area 0:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-area)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospf-area-if)# flood-reduction
```
hello-interval (OSPF)

To specify the interval between consecutive hello packets that are sent on the Open Shortest Path First (OSPF) interface, use the `hello-interval` command in the appropriate mode. To return to the default time, use the `no` form of this command.

```
hello-interval  seconds
no hello-interval
```

**Syntax Description**

- `seconds`: Interval (in seconds). The value must be the same for all nodes on a specific network. Range is 1 to 65535.

**Command Default**

- If this command is not specified in interface configuration mode, then the interface adopts the hello interval parameter specified by the area.
- If this command is not specified in area configuration mode, then the interface adopts the hello interval parameter specified for the process.
- If this command is not specified at any level, then the hello interval is 10 seconds (broadcast) or 30 seconds (non-broadcast).

**Command Modes**

- Interface configuration
- Area configuration
- Router configuration
- Virtual-link configuration
- Multi-area configuration
- Sham-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The hello interval value is advertised in the hello packets. The shorter the hello interval, the faster topological changes are detected, but more routing traffic occurs. This value must be the same for all routers and access servers on a specific network.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the interval between hello packets to 15 seconds:

```
RP/0/RP0/CPU0:router# configure
```
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# hello-interval 15
ignore lsa mospf

To suppress the sending of syslog messages when the router receives link-state advertisement (LSA) Type 6 multicast Open Shortest Path First (MOSPF) packets, which are unsupported, use the `ignore lsa mospf` command in an appropriate configuration mode. To restore the sending of syslog messages, use the `no` form of this command.

```
ignore lsa mospf
no ignore lsa mospf
```

This command has no keywords or arguments.

**Command Default**

When you do not specify this command in router configuration mode, each MOSPF packet received by the router causes the router to send a syslog message.

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Cisco routers do not support LSA Type 6 (MOSPF), and they generate syslog messages if they receive such packets. If the router is receiving many MOSPF packets, you might want to configure the router to ignore the packets and thus prevent a large number of syslog messages.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure the router to suppress the sending of syslog messages when it receives MOSPF packets:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# ignore lsa mospf
```
interface (OSPF)

To define the interfaces on which the Open Shortest Path First (OSPF) protocol runs, use the `interface` command in area configuration mode. To disable OSPF routing for interfaces, use the `interface` form of this command.

```
interface type interface-path-id
no interface type interface-path-id
```

**Syntax Description**

- `type`  Interface type. For more information, use the question mark (?) online help function.
- `interface-path-id`  Physical interface or virtual interface.

**Note**  See the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

**Command Default**

When you do not specify this command in configuration mode, OSPF routing for interfaces is not enabled.

**Command Modes**

Area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `interface` command to associate a specific interface with an area. The interface remains associated with the area even when the IP address of the interface changes.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how the OSPF routing process 109 defines four OSPF areas (0, 2, 3, and 10.9.50.0), and associates an interface with each area:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 4/0/0/3
RP/0/RP0/CPU0:router(config-ospf)# area 2
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospf)# area 3
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 3/0/0/2
```
RP/0/RP0/CPU0:router(config-ospf)# area 10.9.50.0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 3/0/0/1
**log adjacency changes (OSPF)**

To configure the router to send a syslog message when the state of an Open Shortest Path First (OSPF) neighbor changes, use the `log adjacency changes` command in XR Config mode. To turn off this function, use the `disable` keyword. To log all state changes, use the `detail` keyword.

```
log adjacency changes {detail|disable}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>detail</code></td>
<td>Provides all (DOWN, INIT, 2WAY, EXSTART, EXCHANGE, LOADING, FULL) adjacency state changes.</td>
</tr>
<tr>
<td><code>disable</code></td>
<td>Disables sending adjacency change messages.</td>
</tr>
</tbody>
</table>

**Command Default**

The router sends a syslog message when the state of an OSPF neighbor changes.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `log adjacency changes` command to display high-level changes to the state of the peer relationship. Configure this command if you want to know about OSPF neighbor changes.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure the software to send a syslog message for any OSPF neighbor state changes:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# log adjacency changes detail
```
**loopback stub-network**

To enable advertising loopback as stub networks, use the `loopback stub-network` command in an appropriate configuration mode. To disable advertising loopback as stub networks, use the `no` form of this command.

```
loopback stub-network [{enable|disable}]
no loopback stub-network
```

**Syntax Description**
- `enable` (Optional) Enables advertising loopbacks as stub networks.
- `disable` (Optional) Disables advertising loopbacks as stub networks.

**Command Default**
By default, OSPF advertises loopbacks as stub hosts.

**Command Modes**
- OSPF interface configuration
- OSPF router configuration
- OSPF area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
In the interface submode, the command can be enabled only on loopback interfaces.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to enable advertising loopback as a stub network, under OSPF interface configuration:

```
RP/0/RP0/CPU0:router(config)#router ospf 100
RP/0/RP0/CPU0:router(config-ospf)#loopback stub-network enable
```
max-lsa

To limit the number of non-self-generated link-state advertisements (LSAs) that an Open Shortest Path First (OSPF) routing process can keep in the OSPF link-state database (LSDB), use the max-lsa command in XR Config mode. To remove the limit of non self-generated LSAs that an OSPF routing process can keep in the OSPF LSDB, use the no form of this command.

```
max-lsa max [threshold] [warning-only] [ignore-time value] [ignore-count value] [reset-time value]
no max-lsa max [threshold] [warning-only] [ignore-time value] [ignore-count value] [reset-time value]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>max</td>
<td>Maximum number of non-self-generated LSAs the OSPF process can keep in the OSPF LSBD.</td>
</tr>
<tr>
<td>threshold</td>
<td>(Optional) The percentage of the maximum LSA number, as specified by the maximum-number argument, at which a warning message is logged. The default is 75 percent.</td>
</tr>
<tr>
<td>warning-only</td>
<td>(Optional) Specifies that only a warning message is sent when the maximum limit for LSAs is exceeded. Disabled by default.</td>
</tr>
<tr>
<td>ignore-time</td>
<td>(Optional) Specifies the time, in minutes, to ignore all neighbors after the maximum limit of LSAs has been exceeded. The default is 5 minutes.</td>
</tr>
<tr>
<td>ignore-count</td>
<td>(Optional) Specifies the number of times the OSPF process can consecutively be placed into the ignore state. The default is 5 times.</td>
</tr>
<tr>
<td>reset-time</td>
<td>(Optional) Specifies the time, in minutes, after which the ignore count is reset to zero. The default is 2 times ignore-time.</td>
</tr>
</tbody>
</table>

**Command Default**

Disabled

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

This command allows you to protect the OSPF routing process from the large number of received LSAs that can result from a misconfiguration on another router in the OSPF domain (for example, the redistribution of a large number of IP prefixes to OSPF).

When this feature is enabled, the router keeps count of the number of all received (non-self-generated) LSAs. When the configured threshold value is reached, an error message is logged. When the configured max number of received LSAs is exceeded, the router stops accepting new LSAs.

If the count of received LSAs is higher than the configured max number after one minute, the OSPF process disables all adjacencies in the given context and clears the OSPF database. This state is called the ignore state. In this state, all OSPF packets received on all interfaces belonging to the OSPF instance are ignored and no OSPF packets are generated on its interfaces. The OSPF process remains in the ignore state for the duration...
of the configured **ignore-time**. When the **ignore-time** expires, the OSPF process returns to normal operation and starts building adjacencies on all its interfaces.

To prevent the OSPF instance from endlessly oscillating between its normal state and the ignore state, as a result of the LSA count immediately exceeding the **max** number again after it returns from the ignore state, the OSPF instance keeps a count of how many times it has been in the ignore state. This counter is called the **ignore-count**. If the **ignore-count** exceeds its configured value, the OSPF instance remains in the ignore state permanently.

To return the OSPF instance to its normal state, you must issue the **clear ip ospf** command. The **ignore-count** is reset to zero if the LSA count does not exceed the **max** number again during the time configured by the **reset-time** keyword.

If you use the **warning-only** keyword, the OSPF instance never enters the ignore state. When LSA count exceeds the **max** number, the OSPF process logs an error message and the OSPF instance continues in its normal state operation.

### Examples

The following example shows how to configure the OSPF instance to accept 12000 nonself-generated LSAs in the global routing table, and 1000 nonself-generated LSAs in VRF V1.

```bash
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 0
RP/0/RP0/CPU0:router(config-ospf)# max-lsa 12000
RP/0/RP0/CPU0:router(config-ospf)# vrf V1
RP/0/RP0/CPU0:router(config-ospf)# max-lsa 1000
```

The following example shows how to display the current status of the OSPF instance:

```bash
RP/0/RP0/CPU0:router# show ospf 0
Routing Process "ospf 0" with ID 10.0.0.2
NSR (Non-stop routing) is Disabled
Supports only single TOS(TOS0) routes
Supports opaque LSA
It is an area border router
Maximum number of non self-generated LSA allowed 12000
  Current number of non self-generated LSA 1
Threshold for warning message 75%
Ignore-time 5 minutes, reset-time 10 minutes
Ignore-count allowed 5, current ignore-count 0
```
max-metric

To configure the Open Shortest Path First (OSPF) protocol to signal other networking devices not to prefer the local router as an intermediate hop in their shortest path first (SPF) calculations, use the `max-metric` command in XR Config mode. To disable this function, use the `no` form of this command.

```
max-metric router-lsa [external-lsa overriding metric] [include-stub] [on-proc-migration]
[on-proc-restart] [on-startup] [on-switchover] [wait-for-bgp] [summary-lsa]
```

Syntax Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>router-lsa</td>
<td>Always originates router link-state advertisements (LSAs) with the maximum metric.</td>
</tr>
<tr>
<td>external-lsa overriding metric</td>
<td>(Optional) Overrides the external-lsa metric with the max-metric value. The overriding metric argument specifies the number of in-summary-LSAs. The range is 1 to 16777215&gt;. The default is 16711680.</td>
</tr>
<tr>
<td>include-stub</td>
<td>(Optional) Advertises stub links in router-LSA with the max-metric value (0xFFFF).</td>
</tr>
<tr>
<td>on-proc-migration time</td>
<td>(Optional) Sets the maximum metric temporarily after a process migration to originate router-LSAs with the max-metric value. The time range is 5 to 86400 seconds.</td>
</tr>
<tr>
<td>on-proc-restart time</td>
<td>(Optional) Sets the maximum metric temporarily after a process restart to originate router-LSAs with the max-metric value. The time range is 5 to 86400 seconds.</td>
</tr>
<tr>
<td>on-startup time</td>
<td>(Optional) Sets the maximum metric temporarily after a reboot to originate router-LSAs with the max-metric value. The time range is 5 to 86400 seconds.</td>
</tr>
<tr>
<td>on-switchover time</td>
<td>(Optional) Sets the maximum metric temporarily after a switchover to originate router-LSAs with the max-metric value. The time range is 5 to 86400 seconds.</td>
</tr>
<tr>
<td>wait-for-bgp</td>
<td>(Optional) Causes OSPF to originate router LSAs with the maximum metric and allows Border Gateway Protocol (BGP) to decide when to start originating router LSAs with a normal metric instead of the maximum metric.</td>
</tr>
<tr>
<td>summary-lsa</td>
<td>(Optional) specifies the number of in summary-LSAs. The range is 1 to 16777215. The default is 16711680.</td>
</tr>
</tbody>
</table>

**Note**
OSPFWillnotpopulate maximum metric on the router's generated LSAs, when the OSPF routing process is configured to support Nonstop Routing (NSR) or Nonstop Forwarding/Graceful restart (NSF/GR).

**Command Default**
Router LSAs are originated with normal link metrics.

`overriding-metric :16711680`

**Command Modes**
XR Config mode
Use the `max-metric` command to cause the software to originate router LSAs with router link metrics set to LSInfinity (0xFFFF). This feature can be useful in Internet backbone routers that run both OSPF and BGP because OSPF converges more quickly than BGP and may begin attracting traffic before BGP has converged, resulting in dropped traffic.

If this command is configured, the router advertises its locally generated router LSAs with a metric of 0xFFFF. This action allows the router to converge but not attract transit traffic if there are better, alternative paths around this router. After the specified `announce-time` value or notification from BGP has expired, the router advertises the local router LSAs with the normal metric (interface cost).

If this command is configured with the `on-startup` keyword, then the maximum metric is temporarily set only after reboot is initiated. If this command is configured without the `on-startup` keyword, then the maximum metric is permanently used until the configuration is removed.

If the `include-stub` keyword is enabled, the stub-links in the router LSA will be sent with the max-metric. If the `summary-lsa` keyword is enabled, all self-generated summary LSAs will have a metric set to 0xFF0000, unless the metric value is specified with the max-metric value parameter. If the `external-lsa` keyword is enabled, all self-generated external LSAs will have a metric set to 0xFF0000, unless the metric value is specified with the max-metric value parameter.

This command might be useful when you want to connect a router to an OSPF network, but do not want real traffic flowing through it if there are better, alternative paths. If there are no alternative paths, this router still accepts transit traffic as before.

Some cases where this command might be useful are as follows:

- During a router reload, you prefer that OSPF wait for BGP to converge before accepting transit traffic. If there are no alternative paths, the router still accepts transit traffic.
- A router is in critical condition (for example, it has a very high CPU load or does not have enough memory to store all LSAs or build the routing table).
- When you want to gracefully introduce or remove a router to or from the network.
- When you have a test router in a lab, connected to a production network.

For older OSPF implementations (RFC 1247), router links in received router LSAs with a metric and cost of LSInfinity are not used during SPF calculations. Hence, no transit traffic is set to the routers originating such router LSAs.

The following example shows how to configure OSPF to originate router LSAs with the maximum metric until BGP indicates that it has converged:
configure
router ospf 109
max-metric router-LSA on-startup wait-for-bgp
maximum interfaces (OSPF)

To limit the number of interfaces that can be configured for an Open Shortest Path First (OSPF) process, use the `maximum interfaces` command in the appropriate mode. To return to the default limit, use the `no` form of this command.

```
maximum interfaces number-interfaces
no maximum interfaces
```

**Syntax Description**

- **number-interfaces**
  - Number of interfaces. Range is 1 to 4294967295.

**Command Default**

If the command is not specified, the default is 1024.

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `maximum interface` command to increase or decrease the limit on the number of interfaces configured for an OSPF process.

You cannot configure a limit lower than the number of interfaces currently configured for the OSPF process. To lower the limit, remove interfaces from the OSPF configuration until the number of configured interfaces is at or below the desired limit. You may then apply the new, lower limit.

**Task ID**

```
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
```

**Examples**

This example shows how to configure a maximum interface limit of 1500 on a router:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# maximum interfaces 1500
```
maximum paths (OSPF)

To control the maximum number of parallel routes that the Open Shortest Path First (OSPF) protocol can support, use the `maximum paths` command in an appropriate configuration mode. To remove the `maximum paths` command from the configuration file and restore the system to its default condition with respect to the routing protocol, use the `no` form of this command.

```
maximum paths maximum-routes-number
no maximum paths
```

**Syntax Description**
- `maximum-routes-number` Maximum number of parallel routes that OSPF can install in a routing table. Range is 1 to 32.

**Note** The maximum number of paths that can be configured is 32.

**Command Default**
The default value for maximum-paths depends on the platform supported maximum-paths value. 32 paths

**Command Modes**
Router configuration

**Command History**
- Release 6.0 This command was introduced.

**Usage Guidelines**
The default value for maximum-path depends on the platform supported maximum-path value. Any custom value you define for the maximum-path parameter must be within the maximum value supported by platform. The configuration will be rejected if the value you have specified is more then what the platform supports.

When the maximum number of parallel routes is reduced, all existing paths are pruned and paths reinstalled at the new maximum number. During this route-reduction period, you may experience some packet loss for a few seconds. This may impact route traffic.

**Examples**
The following example shows how to allow a maximum of two paths to a destination:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# maximum paths 2
```
maximum redistributed-prefixes (OSPF)

To limit the aggregate number of prefixes that can be redistributed into an Open Shortest Path First (OSPF) process, use the `maximum redistributed-prefixes` command in the appropriate mode. To return to the default limit, use the `no` form of this command.

```
maximum redistributed-prefixes maximum [threshold-value] [warning-only]
no maximum redistributed-prefixes
```

**Syntax Description**

- **maximum**  
  Number of routes. Range is 1 to 4294967295.

- **threshold-value**  
  (Optional) Threshold value (as a percentage) at which to generate a warning message. Range is 1 to 100.

- **warning-only**  
  (Optional) Gives only a warning when the limit is exceeded.

**Command Default**

If the command is not specified, the default is 10000.

The threshold value defaults to 75 percent.

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `maximum redistributed-prefixes` command to increase or decrease the maximum number of prefixes (also referred to as routes) redistributed for an OSPF process.

If the `maximum` value is less than the existing number of routes, existing routes remain configured, but no new routes are redistributed.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure a maximum number of routes that can be redistributed for an OSPF routing process:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# maximum redistributed-prefixes 15000
```
message-digest-key

To specify a key used with Open Shortest Path First (OSPF) Message Digest 5 (MD5) authentication, use the `message-digest-key` command in the appropriate mode. To remove an old MD5 key, use the `no` form of this command.

```
message-digest-key key-id md5 {key|clear key|encrypted key}
no message-digest-key key-id
```

**Syntax Description**

- `key-id` Key number. Range is 1 to 255.
- `md5` Enables OSPF MD5 authentication.
- `key` Alphanumeric string of up to 16 characters.
- `clear` Specifies that the key be clear text.
- `encrypted` Specifies that the key be encrypted using a two-way algorithm.

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the message digest key parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the message digest key parameter specified for the process.

If this command is not specified at any level, then OSPF MD5 authentication is disabled.

**Command Modes**

- Interface configuration
- Area configuration
- XR Config mode
- Virtual-link configuration
- Multi-area configuration
- Sham-link configuration

**Command History**

```
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
```

**Usage Guidelines**

Usually, one key individual interface is used to generate authentication information when packets are sent and to authenticate incoming packets. The same key identifier on the neighbor router must have the same `key` value.

For authentication to be enabled, you must configure the `message-digest-key` command together with the `authentication` command and its `message-digest` keyword. Both the `message-digest-key` and `authentication` commands can be inherited from a higher configuration level.

The process of changing keys is as follows. Suppose the current configuration is:
interface tenGigE 0/3/0/2
message-digest-key 100 md5 OLD

You change the configuration to the following:

interface tenGigE 0/3/0/2
message-digest-key 101 md5 NEW

The system assumes its neighbors do not have the new key yet, so it begins a rollover process. It sends multiple copies of the same packet, each authenticated by different keys. In this example, the system sends out two copies of the same packet—the first one authenticated by key 100 and the second one authenticated by key 101.

Rollover allows neighboring routers to continue communication while the network administrator is updating them with the new key. Rollover stops after the local system finds that all its neighbors know the new key. The system detects that a neighbor has the new key when it receives packets from the neighbor authenticated by the new key.

After all neighbors have been updated with the new key, the old key should be removed. In this example, you would enter the following:

interface ethernet 1
no ospf message-digest-key 100

Then, only key 101 is used for authentication on interface 1.

We recommend that you not keep more than one key individual interface. Every time you add a new key, you should remove the old key to prevent the local system from continuing to communicate with a hostile system that knows the old key. Removing the old key also reduces overhead during rollover.

---

Note

The MD5 key is always stored in encrypted format on the router. The **clear** and **encrypted** keywords inform the router whether the value that is entered is encrypted or unencrypted.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set a new key 19 with the password **8ry4222**:

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# area 0

RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# message-digest-key 19 md5 8ry4222
message-digest-key
mpls ldp auto-config (OSPF)

To enable Label Distribution Protocol (LDP)-Interior Gateway Protocol (IGP) interface automatic configuration, use the `mpls ldp auto-config` command in the appropriate mode. To disable LDP-IGP interface automatic configuration, use the `no` form of this command.

```
mpls ldp auto-config
no mpls ldp auto-config
```

**Command Default**

LDP-IGP interface automatic configuration is disabled for OSPF.

**Command Modes**

- Interface configuration
- Area configuration
- XR Config mode

**Command History**

```
Release   Modification
---------  ------------
Release 6.0  This command was introduced.
```

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

```
Task  Operations
ID    
ospf  read, write
```

**Examples**

The following example shows how to enable LDP-IGP interface automatic configuration:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 10
RP/0/RP0/CPU0:router(config-ospf)# mpls ldp auto-config
```
mpls ldp sync (OSPF)

To enable Label Distribution Protocol (LDP)-Interior Gateway Protocol (IGP) synchronization, use the `mpls ldp sync` command in the appropriate mode. To disable LDP-IGP synchronization, use the `no` form of this command.

```
mpls ldp sync [disable]
no mpls ldp sync
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disable (Optional) Disables MPLS LDP synchronization from within the OSPF interface and area configuration submodes only. For the OSPF router configuration mode, use the <code>no</code> form of the command.</td>
</tr>
</tbody>
</table>

**Command Default**

LDP-IGP synchronization is disabled for OSPF.

**Command Modes**

- Interface configuration
- Area configuration
- XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to enable LDP-IGP synchronization:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# mpls ldp sync
```
**mtu-ignore (OSPF)**

To prevent Open Shortest Path First (OSPF) from checking whether neighbors are using the same maximum transmission unit (MTU) on a common interface when exchanging database descriptor (DBD) packets, use the `mtu-ignore` command in the appropriate mode. To reset to default, use the `no` form of this command.

```
mtu-ignore [{disable|enable}]
no mtu-ignore
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>disable</code></td>
<td>(Optional) Enables checking for whether OSPF neighbors are using the MTU on a common interface.</td>
</tr>
<tr>
<td><code>enable</code></td>
<td>(Optional) Disables checking for whether OSPF neighbors are using the MTU on a common interface.</td>
</tr>
</tbody>
</table>

**Command Default**

The default is `mtu-ignore` with no keywords, which disables MTU checking.

If this command is not specified in interface configuration mode, then the interface adopts the MTU ignore parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the MTU ignore parameter specified for the process.

If this command is not specified at any level, then OSPF checks the MTU received from neighbors when exchanging DBD packets.

**Command Modes**

- Interface configuration
- Area configuration
- XR Config mode
- Multi-area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

OSPF checks whether OSPF neighbors are using the same MTU on a common interface. This check is performed when neighbors exchange DBD packets. If the receiving MTU in the DBD packet is higher than the MTU configured on the incoming interface, OSPF adjacency is not established.

The keywords, `disable` and `enable`, do not need to be used. If no keywords are used, the `mtu-ignore` command disables MTU checking. You can then use the `no mtu-ignore` command to activate MTU checking.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to disable MTU mismatch detection on receiving DBD packets:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospf-ar-if)# mtu-ignore
```
multi-area-interface

To enable multiple adjacencies for different Open Shortest Path First (OSPF) areas and enter multi-area interface configuration mode, use the `multi-area-interface` command in the area configuration mode. To reset to the default, use the `no` form of this command.

```
multi-area-interface type interface-path-id
no multi-area-interface type interface-path-id
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>type</code></td>
<td>Interface type. For more information, use the question mark (?) online help function.</td>
</tr>
<tr>
<td><code>interface-path-id</code></td>
<td>Physical interface or virtual interface.</td>
</tr>
</tbody>
</table>

**Note**

Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

**Command Default**

An OSPF network is enabled for one area only.

**Command Modes**

Area configuration

**Command History**

```
Release 6.0  This command was introduced.
```

**Usage Guidelines**

Use the `multi-area-interface` command to enable area border routers (ABRs) to establish multiple adjacencies for different OSPF areas.

Each multiple area adjacency is announced as a point-to-point unnumbered link in the configured area. This point-to-point link provides a topological path for that area. The first or primary adjacency using the link advertises the link consistent with draft-ietf-ospf-multi-area-adj-06.txt.

You can configure multi-area adjacency on any interface where only two OSPF speakers are attached. In the case of native broadcast networks, the interface must be configured as an OSPF point-to-point type using the `network point-to-point` command to enable the interface for a multi-area adjacency.

**Task ID**

```
Task ID  Operations
ospf     read,
         write
```

**Examples**

The following example shows how to enable multiple area adjacency for OSPF 109:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# area 0
```
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospf-ar-if)# area 1
RP/0/RP0/CPU0:router(config-ospf-ar)# multi-area-interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospf-ar-mif)# ?

- **authentication**: Enable authentication
- **authentication-key**: Authentication password (key)
- **commit**: Commit the configuration changes to running
- **cost**: Interface cost
- **database-filter**: Filter OSPF LSA during synchronization and flooding
- **dead-interval**: Interval after which a neighbor is declared dead
- **describe**: Describe a command without taking real actions
- **distribute-list**: Filter networks in routing updates
- **do**: Run an exec command
- **exit**: Exit from this submode
- **hello-interval**: Time between HELLO packets
- **message-digest-key**: Message digest authentication password (key)
- **mtu-ignore**: Enable/Disable ignoring of MTU in DBD packets
- **no**: Negate a command or set its defaults
- **packet-size**: Customize size of OSPF packets upto MTU
- **pwd**: Commands used to reach current submode
- **retransmit-interval**: Time between retransmitting lost link state advertisements
- **root**: Exit to the global configuration mode
- **show**: Show contents of configuration
- **transmit-delay**: Estimated time needed to send link-state update packet
neighbor (OSPF)

To configure Open Shortest Path First (OSPF) routers interconnecting to nonbroadcast networks, use the `neighbor` command in interface configuration mode. To remove a configuration, use the `no` form of this command.

```
neighbor ip-address [cost number] [priority number] [poll-interval seconds]
no neighbor ip-address [cost number] [priority number] [poll-interval seconds]
```

### Syntax Description

- **ip-address**: Interface IP address of the neighbor.
- **cost number** *(Optional)*: Assigns a cost to the neighbor, in the form of an integer from 1 to 65535. Neighbors with no specific cost configured assume the cost of the interface, based on the `cost` command. On point-to-multipoint interfaces, `cost number` is the only keyword and argument combination that works. The `cost` keyword does not apply to nonbroadcast multiaccess (NBMA) networks.
- **priority number** *(Optional)*: Specifies an 8-bit number indicating the router priority value of the nonbroadcast neighbor associated with the IP address specified. The `priority` keyword does not apply to point-to-multipoint interfaces.
- **poll-interval seconds** *(Optional)*: Specifies an unsigned integer value (in seconds) reflecting the poll interval. RFC 1247 recommends that this value be much larger than the hello interval. The `poll-interval` keyword does not apply to point-to-multipoint interfaces.

### Command Default

No configuration is specified.

- **priority number**: 0
- **poll-interval seconds**: 120 seconds (2 minutes)

### Command Modes

Interface configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced</td>
</tr>
</tbody>
</table>

### Usage Guidelines

You must include one neighbor entry in the software configuration for each known nonbroadcast network neighbor. The neighbor address must be on the primary address of the interface.

If a neighboring router has become inactive (hello packets have not been received for the router dead interval period), it may still be necessary to send hello packets to the dead neighbor. These hello packets are sent at a reduced rate called the `poll interval`.

When the router starts up, it sends only hello packets to those routers with nonzero priority; that is, routers that are eligible to become designated routers (DRs) and backup designated routers (BDRs). After the DR and BDR are selected, the DR and BDR start sending hello packets to all neighbors to form adjacencies.

To filter all outgoing OSPF link-state advertisement (LSA) packets for the neighbor, use the `neighbor database-filter all out` command.
The following example shows how to declare a router at address 172.16.3.4 on a nonbroadcast network, with a priority of 1 and a poll interval of 180 seconds:

```
RP/0/RP0/CPU0:router(config-ospf-ar-if)# neighbor 172.16.3.4 priority 1 poll-interval 180
```

The following example illustrates a network with nonbroadcast:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tenGigE 1/0/0/3
RP/0/RP0/CPU0:router(config-if)# ip address 172.16.3.10 255.255.255.0

RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf)# interface tenGigE 1/0/0/3
RP/0/RP0/CPU0:router(config-ospf-if)# network nonbroadcast
RP/0/RP0/CPU0:router(config-ospf-if)# neighbor 172.16.3.4 priority 1 poll-interval 180

RP/0/RP0/CPU0:router(config-ospf-if)# neighbor 172.16.3.5 cost 10 priority 1 poll-interval 180
RP/0/RP0/CPU0:router(config-ospf-if)# neighbor 172.16.3.6 cost 15 priority 1 poll-interval 180
RP/0/RP0/CPU0:router(config-ospf-if)# neighbor 172.16.3.7 priority 1 poll-interval 180
```
neighbor database-filter all out

To filter all outgoing link-state advertisements (LSAs) to an Open Shortest Path First (OSPF) neighbor, use the `neighbor database-filter all out` command in interface configuration mode. To restore the forwarding of LSAs to the neighbor, use the `no` form of this command.

```
neighbor ip-address database-filter all out
no neighbor ip-address database-filter all out
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>ip-address</code></td>
<td>IP address of the neighbor to which outgoing LSAs are blocked.</td>
</tr>
</tbody>
</table>

**Command Default**

Instead of all outgoing LSAs being filtered to the neighbor, they are flooded to the neighbor.

**Command Modes**

Interface configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `neighbor database-filter all out` command to filter all outgoing OSPF LSA packets during synchronization and flooding for point-to-multipoint neighbors on nonbroadcast networks. More neighbor options are available with the `neighbor` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to prevent flooding of OSPF LSAs from point-to-multipoint networks to the neighbor at IP address 10.2.3.4:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 1/0/0/3
RP/0/RP0/CPU0:router(config-ospf-ar-if)# neighbor 10.2.3.4 database-filter all out
```
network (OSPF)

To configure the Open Shortest Path First (OSPF) network type to a type other than the default for a given medium, use the **network** command in the appropriate mode. To return to the default value, use the **no** form of this command.

```
network {broadcast|non-broadcast|[point-to-multipoint [non-broadcast]|point-to-point]}
no network
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>broadcast</td>
<td>Sets the network type to broadcast.</td>
</tr>
<tr>
<td>non-broadcast</td>
<td>Sets the network type to nonbroadcast multiaccess (NBMA).</td>
</tr>
<tr>
<td>point-to-multipoint</td>
<td>Sets the network type to point-to-multipoint.</td>
</tr>
<tr>
<td>non-broadcast</td>
<td>(Optional) Sets the point-to-multipoint network to be nonbroadcast. If you use this keyword, the <strong>neighbor</strong> command is required.</td>
</tr>
<tr>
<td>point-to-point</td>
<td>Sets the network type to point-to-point.</td>
</tr>
</tbody>
</table>

### Command Default

If this command is not specified in interface configuration mode, then the interface adopts the network parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the network parameter specified for the process.

If this command is not specified at any level, then the OSPF network type is the default of the given medium.

### Command Modes

- Interface configuration
- Area configuration
- Router configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the **network** command to configure broadcast networks as NBMA networks when, for example, routers in your network do not support multicast addressing.

Configuring NBMA networks as either broadcast or nonbroadcast assumes that there are virtual circuits from every router to every router or fully meshed network. However, there are other configurations where this assumption is not true; for example, a partially meshed network. In these cases, you can configure the OSPF network type as a point-to-multipoint network. Routing between two routers that are not directly connected go through the router that has virtual circuits to both routers. You need not configure neighbors when using this command.

If this command is issued on an interface that does not allow it, this command is ignored.

OSPF has two features related to point-to-multipoint networks. One feature applies to broadcast networks; the other feature applies to nonbroadcast networks:
On point-to-multipoint, broadcast networks, you can use the `neighbor` command, and you must specify a cost to that neighbor.

On point-to-multipoint, nonbroadcast networks, you must use the `neighbor` command to identify neighbors. Assigning a cost to a neighbor is optional.

### Examples

The following example shows how to configure the OSPF network as a nonbroadcast network:

```plaintext
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospf-ar-if)# network non-broadcast
RP/0/RP0/CPU0:router(config-ospf-ar-if)# neighbor 172.16.3.4 priority 1 poll-interval 180
```
To configure nonstop forwarding (NSF) for the Open Shortest Path First (OSPF) protocol, use the `nsf` command in the appropriate mode. To remove this command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
nsf {cisco [enforce global]|ietf [helper disable]}
no nsf {cisco [enforce global]|ietf [helper disable]}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>cisco</code></td>
<td>Enables Cisco Nonstop Forwarding.</td>
</tr>
<tr>
<td><code>enforce global</code></td>
<td>(Optional) Cancels NSF restart when non-NSF network device neighbors are detected.</td>
</tr>
<tr>
<td><code>ietf</code></td>
<td>Enables Internet Engineering Task Force (IETF) graceful restart.</td>
</tr>
<tr>
<td><code>helper disable</code></td>
<td>(Optional) Disables router helper support.</td>
</tr>
</tbody>
</table>

### Command Default

NSF is disabled.

### Command Modes

XR Config mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

The NSF feature allows for the forwarding of data packets to continue along known routes while routing protocol information (such as OSPF) is being restored following a switchover.

Use the `nsf` command if the router is expected to perform NSF during restart. To experience the full benefits of this feature, configure all neighboring routers with NSF.

When this command is used without the optional `cisco enforce global` keywords and non-NSF neighbors are detected, the NSF restart mechanism aborts on the interfaces of those neighbors and functions properly on others.

When this command is used with the optional `cisco enforce global` keywords and non-NSF neighbors are detected, NSF restart is canceled for the entire OSPF process.

IETF graceful restart provides an NSF mechanism to allow data traffic to flow seamlessly with no packet drops during the transient period when OSPF attempts to recover after a process restart or RP failover, within the guidelines of RFC 3623.

By default, neighbors in helper mode listen to both the NSF Cisco- and NSF IETF-type LSAs. The `nsf` command enables one type of mechanism that would undergo an RP failover or, anticipating an OSPF process restart. If the `cisco` or `ietf` keyword is not entered, NSF is not enabled, irrespective of neighbors in listening mode for both NSF Cisco and NSF IETF.
The following example shows how to cancel NSF restart for the entire OSPF process if non-NSF neighbors are detected on any network interface during restart:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# nsf cisco enforce global
```
**nsf flush-delay-time (OSPF)**

To configure the maximum time allowed for nonstop forwarding (NSF) external route queries for the Open Shortest Path First (OSPF) protocol, use the `nsf flush-delay-time` command in the appropriate mode. To remove this command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
nsf flush-delay-time seconds
no nsf flush-delay-time seconds
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>seconds</code></td>
<td>Length of time (in seconds) allowed for NSF external route queries. Range is 1 to 3600 seconds.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command Default</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>seconds</code> : 300</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command Modes</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router configuration</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command History</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Usage Guidelines</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>No specific guidelines impact the use of this command.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task ID</td>
<td>Operations</td>
</tr>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure the maximum time for NSF to learn external routes for OSPF at 60 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# nsf flush-delay-time 60
```
nsf interval (OSPF)

To configure the minimum time between consecutive nonstop forwarding (NSF) restart attempts for the Open Shortest Path First (OSPF) protocol, use the `nsf interval` command in the appropriate mode. To remove this command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
nsf interval seconds
no nsf interval seconds
```

**Syntax Description**
- `seconds`: Length of time (in seconds) between consecutive restart attempts. Range is 90 to 3600 seconds.

**Command Default**
- `seconds`: 90

**Command Modes**
- Router configuration

**Command History**
- Release 6.0: This command was introduced.

**Usage Guidelines**
When you use the `nsf interval` command, the OSPF process must be up for at least 90 seconds before OSPF attempts to perform an NSF restart.

**Task ID**
- `ospf` read, write

**Examples**
The following example shows how to configure the minimum time between consecutive NSF restart attempts at 120 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# nsf interval 120
```
nsf lifetime (OSPF)

To configure the maximum time that routes are held in the Routing Information Base (RIB) following an Open Shortest Path First (OSPF) process restart, use the `nsf lifetime` command in the appropriate mode. To remove this command from the configuration file and restore the system to its default condition, use the `no` form of this command.

```
nsf lifetime seconds
no nsf lifetime seconds
```

**Syntax Description**

- `seconds`: The length of time (in seconds) that routes are held in the RIB. Range is 90 to 3600 seconds.

**Command Default**

`seconds`: 95

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When you use this command, the OSPF process must reconverge within the maximum length of time configured. If the convergence exceeds this length of time, routes are purged from RIB and nonstop forwarding (NSF) restart may fail.

**Task ID**

- **OSPF**
  - `read`
  - `write`

**Examples**

The following example shows how to configure the maximum lifetime for OSPF NSF at 120 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# nsf lifetime 120
```
nssa (OSPF)

To configure an area as a not-so-stubby area (NSSA), use the `nssa` command in area configuration mode. To remove the NSSA distinction from the area, use the `no` form of this command.

```
nssa [no-redistribution] [default-information-originate [metric metric-value] [metric-type type-value]] [no-summary]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>no-redistribution</td>
<td>(Optional) Imports routes only into the normal areas, but not into the NSSA area, by the <code>redistribute</code> command when the router is an NSSA Area Border Router (ABR).</td>
</tr>
<tr>
<td>default-information-originate</td>
<td>(Optional) Generates a Type 7 default into the NSSA area. This keyword takes effect only on an NSSA ABR or NSSA Autonomous System Boundary Router (ASBR).</td>
</tr>
<tr>
<td>metric metric-value</td>
<td>(Optional) Specifies the metric used for generating the default route. If you omit a value and do not specify a value using the <code>defaultmetric</code> command, the default metric value is 10. Range is 1 to 16777214.</td>
</tr>
<tr>
<td>metric-type type-value</td>
<td>(Optional) Specifies the external link type associated with the default route advertised into the OSPF routing domain. It can be one of the following values:</td>
</tr>
<tr>
<td></td>
<td>1—Type 1 external route</td>
</tr>
<tr>
<td></td>
<td>2—Type 2 external route</td>
</tr>
<tr>
<td>no-summary</td>
<td>(Optional) Prevents an ABR from sending summary link advertisements into the NSSA.</td>
</tr>
</tbody>
</table>

### Command Default

No NSSA area is defined.

### Command Modes

Area configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

An NSSA does not flood Type 5 external LSAs from the core into the area, but can import autonomous system external routes in a limited fashion within the area.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to configure area 1 as an NSSA area:
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 1
RP/0/RP0/CPU0:router(config-ospf-ar)# nssa
ospf name-lookup

To configure the Open Shortest Path First (OSPF) protocol to look up Domain Name System (DNS) names, use the `ospf name-lookup` command in XR Config mode. To disable this function, use the `no` form of this command.

```
ospf name-lookup
no ospf name-lookup
```

**Command Default**

Routers are displayed by router ID or neighbor ID.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `ospf name-lookup` command to easily identify a router when executing all OSPF `show` command displays. The router is displayed by name rather than by its router ID or neighbor ID.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure OSPF to identify a router by name:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ospf name-lookup
```
packet-size (OSPF)

To configure the size of Open Shortest Path First (OSPF) packets up to the size specified by the maximum transmission unit (MTU), use the `packet-size` command in the appropriate configuration mode. To disable this function and reestablish the default packet size, use the `no` form of this command.

```
packet-size bytes
no packet-size
```

**Syntax Description**

- `bytes` Size, in bytes. Range is 576 to 10000 bytes.

**Command Default**

If the command is not specified, the default packet size is either the interface IP MTU size (if that is lower than 9000 bytes) or 9000 bytes.

**Command Modes**

- XR Config mode
- Area configuration
- Interface configuration
- Multi-area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `packet-size` command to customize the size of OSPF packets. The OSPF protocol compares the packet size and the MTU size and uses the lower packet size value.

If the command is not configured, the default packet size is equal to the interface IP MTU size (if that is lower than 9000 bytes) or 9000 bytes. For example, if the interface IP MTU size is 1500 bytes, OSPF uses packet size of 1500 bytes on the interface because the byte size is lower than 9000 bytes. If the interface IP MTU size is 9500 bytes, OSPF uses packet size of 9000 bytes on the interface because the byte size exceeds 9000 bytes. The interface IP MTU size depends on the interface and the platform. In most cases, the default interface IP MTU value will be lower than 9000 bytes.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure the packet size on an interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 1/0/0/2
```
RP/0/RP0/CPU0:router(config-ospf-ar-if)# packet-size 3500
passive (OSPF)

To suppress the sending of Open Shortest Path First (OSPF) protocol operation on an interface, use the **passive** command in the appropriate mode. To remove the passive configuration, use the **no** form of this command.

```
passive [ { disable | enable } ]
no passive
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disable</td>
<td>(Optional) Sends OSPF updates.</td>
</tr>
<tr>
<td>enable</td>
<td>(Optional) Disables sending OSPF updates.</td>
</tr>
</tbody>
</table>

**Command Default**

- If this command is not specified in interface configuration mode, then the interface adopts the passive parameter specified by the area.
- If this command is not specified in area configuration mode, then the interface adopts the passive parameter specified for the process.
- If this command is not specified at any level, then the passive parameter is disabled and OSPF updates are sent on the interface.

**Command Modes**

- Interface configuration
- Area configuration
- XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

OSPF routing information is neither sent nor received through the specified interface. The interface appears as a stub network in the OSPF router (Type 1) link-state advertisement (LSA).

**Examples**

The following example shows that GigabitEthernet interface 1/0/0/2 reduces OSPF updates because passive mode is enabled; however, tenGigE interface 0/1/0/3 receives normal OSPF traffic flow:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 1/0/0/2
RP/0/RP0/CPU0:router(config-ospf-ar-if)# passive
RP/0/RP0/CPU0:router(config-ospf-ar-if)# exit
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 1/0/0/3
```
RP/0/RP0/CPU0:router(config-ospf-ar-if)# end
priority (OSPF)

To set the router priority for an interface, which helps determine the designated router for an Open Shortest Path First (OSPF) link, use the `priority` command in the appropriate mode. To return to the default value, use the `no` form of this command.

```
priority value
no priority value
```

**Syntax Description**
- `value` 8-bit unsigned integer indicating the router priority value. Range is 0 to 255.

**Command Default**
- If this command is not specified in interface configuration mode, then the interface adopts the priority parameter specified by the area.
- If this command is not specified in area configuration mode, then the interface adopts the priority parameter specified for the process.
- If this command is not specified at any level, then the default priority is 1.

**Command Modes**
- Interface configuration
- Area configuration
- Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When two routers attached to a network both attempt to become the designated router, the one with the higher router priority takes precedence. If there is a tie, the router with the higher router ID takes precedence. A router with a router priority set to zero is ineligible to become the designated router or backup designated router. Router priority is configured only for interfaces to multiaccess networks (in other words, not point-to-point networks).

This priority value is used when you configure the Open Shortest Path First (OSPF) protocol for nonbroadcast networks using the `neighbor` command for OSPF.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows that priority is set through the `priority` and `neighbor` commands for Routers A and B and that the neighbor priority value must reflect that of the neighbor router:

```
Router A
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tenGigE 0/1/0/1
```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.0.0.2 255.255.255.0
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# network non-broadcast
RP/0/RP0/CPU0:router(config-ospf-ar-if)# priority 4
RP/0/RP0/CPU0:router(config-ospf-ar-if)# neighbor 10.0.0.1 priority 6

Router B
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tenGigE POS 0/2/0/1
RP/0/RP0/CPU0:router(config-if)# ipv4 address 10.0.0.1 255.255.255.0
RP/0/RP0/CPU0:router(config-if)# exit
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# network non-broadcast
RP/0/RP0/CPU0:router(config-ospf-ar-if)# priority 6
RP/0/RP0/CPU0:router(config-ospf-ar-if)# neighbor 10.0.0.2 priority 4
**protocol shutdown**

To disable an instance of the Open Shortest Path First (OSPF) protocol so that it cannot form an adjacency on any interface, use the `protocol shutdown` command in the XR Config mode. To reenable the OSPF protocol, use the `no` form of this command.

```
protocol shutdown
no protocol shutdown
```

**Command Default**

No default behavior or values

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `protocol shutdown` command to disable the OSPF protocol for a specific routing instance without removing any existing OSPF configuration parameters.

The OSPF protocol continues to run on the router and you can use the current OSPF configuration, but OSPF does not form any adjacencies on any interface.

This command is similar to performing the `no router ospf` command.

**Task ID**

```
Task ID  Operations
ospf     read, write
```

**Examples**

The following example shows how to disable the OSPF 1 instance:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospf)# protocol shutdown
```
queue dispatch flush-lsa

To change the number of LSAs scheduled (rate-limited) for flushing, that are processed in each iteration, use the `queue dispatch flush-lsa` command in XR Config mode. To return to the system default value, use the `no` form of this command.

```
queue dispatch flush-lsa \( count \)
no queue dispatch flush-lsa
```

**Syntax Description**

- `count`: Maximum number of LSAs flushed per run. Range is 30 to 3000.

**Command Default**

The default LSAs flushed per run is 150 (when the count is not configured).

**Command Modes**

XR Config mode

**Command History**

- **Release 6.0**: This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- `ospf`: read, write

**Examples**

The following example shows how to limit the number of LSAs flushed per run to 30:

```
RP/0/RP0/CPU0:router(config-ospf)# queue dispatch flush-lsa 30
```

Use the `show ospf message-queue, on page 276` command to see the queue dispatch values, peak lengths, and limits.
queue dispatch incoming

To limit the number of incoming packets (LSAUpdates, LSAck, DBDs, LSRequests, and Hellos that trigger a change state) processed, use the `queue dispatch incoming` command in XR Config mode. To return to the system default value, use the `no` form of this command.

```
queue dispatch incoming count
no queue dispatch incoming
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>Maximum number of continuous events processed. Range is 30 to 3000.</td>
</tr>
</tbody>
</table>

**Command Default**

The default incoming count is 300 packets (when the count is not configured).

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how limit the number of incoming packets processed to 500:

```
RP/0/RP0/CPU0:router(config-ospf)# queue dispatch incoming 500
```

Use the `show ospf message-queue`, on page 276 command to see the queue dispatch values, peak lengths, and limits.
queue dispatch rate-limited-lsa

To set the maximum number of rate-limited link-state advertisement (LSA) (re-)originations processed per run, use the `queue dispatch rate-limited-lsa` command in XR Config mode. To return to the system default value, use the `no` form of this command.

```
queue dispatch rate-limited-lsa count
no queue dispatch rate-limited-lsa
```

**Syntax Description**
- `count`  Maximum number of rate-limited LSAs processed per run. Range is 30 to 3000.

**Command Default**
The default number of rate-limited LSAs processed per run is 300 (when this count is not configured).

**Command Modes**
XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to set the maximum number of rate-limited LSA (re-)originations processed per run to 300:

```
RP/0/RP0/CPU0:router(config-ospf)# queue dispatch rate-limited-lsa 300
```
queue dispatch spf-lsa-limit

To change the maximum number of Type 3-4 and Type 5-7 link-state advertisements (LSAs) processed per shortest path first (SPF) iteration within a single SPF run, use the `queue dispatch spf-lsa-limit` command in XR Config mode. To return to the system default value, use the `no` form of this command.

```
queue dispatch spf-lsa-limit count
no queue dispatch spf-lsa-limit
```

**Syntax Description**

- `count` Maximum number of continuous Type 3-4 and Type 5-7 LSAs processed per SPF in each scheduled iteration within a single SPF run. Range is 30 to 3000.

**Command Default**

The default number of Type 3-4 and Type 5-7 processed per run is 150 LSAs (when this command is not configured).

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- **ospf** read, write

**Examples**

The following example shows how to limit the number of continuous Type 3-4 and Type 5-7 LSAs processed by SPF per scheduling run, to 100:

```
RP/0/RP0/CPU0:router(config-ospf)# queue dispatch spf-lsa-limit 100
```
queue limit

To set the high watermark for incoming events by priority, use the `queue limit` in XR Config mode. To return to the system default values, use the `no` form of this command.

```
queue limit {high|medium|low} count
no queue limit {high|medium|low}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>high</code></td>
<td>High watermark for incoming high-priority events (state-changing Hellos).</td>
</tr>
<tr>
<td><code>medium</code></td>
<td>High watermark for incoming medium-priority events (LSA ACK).</td>
</tr>
<tr>
<td><code>low</code></td>
<td>High watermark for incoming low-priority events (DBD/LSUpd/LSReq).</td>
</tr>
<tr>
<td><code>count</code></td>
<td>Maximum number of events per queue. Events are dropped when the priority queue size exceeds this value. Range is 1000 to 30000.</td>
</tr>
</tbody>
</table>

**Command Default**

High watermark: 9500 (when the corresponding configuration is not present).

Medium watermark: 9000 (when the corresponding configuration is not present).

Low watermark: 8000 (when the corresponding configuration is not present).

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Always keep the limits in the following order of priority:

Limit for High > Limit for Medium > Limit for Low

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following examples show how to set the maximum number of events per queue:

```
RP/0/RP0/CPU0:router(config-ospf)# queue limit high 11000
RP/0/RP0/CPU0:router(config-ospf)# queue limit medium 10000
RP/0/RP0/CPU0:router(config-ospf)# queue limit low 9000
```
range (OSPF)

To consolidate and summarize routes at an area boundary, use the range command in area configuration mode. To disable this function, use the no form of this command.

```
range ip-address mask [{advertise|not-advertise}]
no range ip-address mask [{advertise|not-advertise}]
```

**Syntax Description**
- `ip-address` IP address in four-part, dotted-decimal notation.
- `mask` IP address mask.
- `advertise` (Optional) Sets the address range status to advertise and generates a Type 3 summary link-state advertisement (LSA).
- `not-advertise` (Optional) Sets the address range status to DoNotAdvertise. The Type 3 summary LSA is suppressed and the component networks remain hidden from other networks.

**Command Default**
When this command is not specified for Area Border Routers (ABRs), routes at an area boundary are not consolidated or summarized.
Advertise is the default.

**Command Modes**
Area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the range command only with Area Border Router (ABRs). Use the command to consolidate or summarize routes for an area. The result is that a single summary route is advertised to other areas by the ABR. Routing information is condensed at area boundaries. External to the area, a single route is advertised for each address range. This process is called route summarization.

Multiple range configurations specifying the range command can be configured. Thus, the OSPF protocol can summarize addresses for many different sets of address ranges.

The summarized route uses the maximum cost of the routes assumed in the range.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows area 36.0.0.0 consisting of interfaces whose IP addresses have “10.31.x.x” as the first two octets. The range command summarizes interfaces. Instead of advertising eight networks individually, the single route 10.31.0.0 255.255.0.0 is advertised:
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar-if)# interface tenGigE 0/3/0/2
! RP/0/RP0/CPU0:router(config-ospf)# area 36.0.0.0
RP/0/RP0/CPU0:router(config-ospf-ar)# range 10.31.0.0 255.255.0.0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/0
RP/0/RP0/CPU0:router(config-ospf-ar-if)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# interface tenGigE 0/1/0/2
RP/0/RP0/CPU0:router(config-ospf-ar-if)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospf-ar-if)# interface tenGigE 0/2/0/0
RP/0/RP0/CPU0:router(config-ospf-ar-if)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# interface tenGigE 0/2/0/2
RP/0/RP0/CPU0:router(config-ospf-ar-if)# interface tenGigE 0/2/0/3
RP/0/RP0/CPU0:router(config-ospf-ar-if)# end
redistribute (OSPF)

To redistribute routes from one routing domain into Open Shortest Path First (OSPF), use the `redistribute` command in the appropriate mode. To remove the `redistribute` command from the configuration file and restore the system to its default condition in which the software does not redistribute routes, use the `no` form of this command.

**Border Gateway Protocol (BGP)**
```
redistribute bgp process-id [preserve-med] [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```
```
no redistribute bgp process-id [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```

**Local Interface Routes**
```
redistribute connected [instance instance-name] [instance IPCP] [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```
```
no redistribute connected [instance instance-name] [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```

**Directed-attached gateway redundancy (DAGR)**
```
redistribute dagr [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```
```
no redistribute dagr [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```

**Intermediate System-to-Intermediate System (IS-IS)**
```
redistribute isis process-id [{level-1|level-2|level-1-2}] [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```
```
no redistribute isis process-id [{level-1|level-2|level-1-2}] [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```

**Open Shortest Path First (OSPF)**
```
redistribute ospf process-id [match {external [{1|2}]|internal|nssa-external [{1|2}]|] [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```
```
no redistribute ospf process-id [match {external [{1|2}]|internal|nssa-external [{1|2}]|] [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```

**IP Static Routes**
```
redistribute static [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```
```
no redistribute static [metric metric-value] [metric-type {1|2}] [route-policy policy-name] [tag tag-value]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgp</td>
<td>Distributes routes from the BGP protocol.</td>
</tr>
</tbody>
</table>

Routing Command Reference for Cisco NCS 5000 Series Routers
### process-id
For the `bgp` keyword, an autonomous system number has the following ranges:

- Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.
- Range for 4-byte Autonomous system numbers (ASNs) in as plain format is 1 to 4294967295.
- Range for 4-byte Autonomous system numbers (ASNs) as a dot format is 1.0 to 65535.65535.

For the `isis` keyword, an IS-IS instance name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.

For the `ospf` keyword, an OSPF instance name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.

### preserve-med
(Optional) Preserves the Multi Exit Discriminator (MED) of BGP routes.

### metric metric-value
(Optional) Specifies the metric used for the redistributed route. Range is 1 to 16777214. Use a value consistent with the source protocol.

### metric-type { 1 | 2 }
(Optional) Specifies the external link type associated with the route advertised into the OSPF routing domain. It can be one of two values:

- **1** — Type 1 external route
- **2** — Type 2 external route

### tag tag-value
(Optional) Specifies the value attached to each external route. This value is not used by the OSPF protocol itself, but is carried in the external LSAs. Range is 0 to 4294967295.

### route-policy policy-name
(Optional) Specifies the identifier of a configured policy. A policy is used to filter the importation of routes from this source routing protocol to OSPF.

### connected
Distributes routes that are established automatically by virtue of having enabled IP on an interface.

### instance
Connected instance.

### instance-name
Name of the connected instance.

### instance IPCP
Distributes routes from IPCP protocols.

### isis
Distributes routes from the IS-IS protocol.

### level-1
(Optional) Redistributes Level 1 routes into other IP routing protocols independently.

### level-1-2
(Optional) Distributes both Level 1 and Level 2 routes into other IP routing protocols.

### level-2
(Optional) Distributes Level 2 routes into other IP routing protocols independently.

### ospf
Distributes routes from the OSPF protocol.
match &{ internal | external [1 | 2] | nssa-external [1 | 2] }

(Optional) Specifies the criteria by which OSPF routes are redistributed into other routing domains. It can be one or more of the following:

- **internal** — Routes that are internal to a specific autonomous system (intra- and inter-area OSPF routes).
- **external [1 | 2]** — Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 external routes.
- **nssa-external [1 | 2]** — Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 not-so-stubby area (NSSA) external routes.

For the **external** and **nssa-external** options, if a type is not specified, then both Type 1 and Type 2 are assumed.

If no match is specified, the default is no filtering.

### Command Default

Route redistribution is disabled.

**metric metric-value**: Default is 20 for routes from all protocols except BGP routes, for which the default is 1.

**metric-type**: Type 2 external route.

### Command Modes

Router configuration

### Command History

**Release 6.0**: This command was introduced.

### Usage Guidelines

When redistributing routes (into OSPF) using both command keywords for setting or matching of attributes and a route policy, the routes are run through the route policy first, followed by the keyword matching and setting.

Redistributed routing information should always be filtered by the **policy policy-name** keyword and argument. This filtering ensures that only those routes intended by the administrator are redistributed into OSPF.

For information about routing policies, see the *Routing Policy Commands on Router Routing Command Reference Guide*.

Whenever you use the **redistribute** or **default-information originate (OSPF)**, on page 178 command to redistribute routes into an OSPF routing domain, the router automatically becomes an ASBR. However, an ASBR does not, by default, generate a default route into the OSPF routing domain.

When routes are redistributed between OSPF processes, no OSPF metrics are preserved.
When routes are redistributed into OSPF and no metric is specified with the **metric** keyword, OSPF uses 20 as the default metric for routes from all protocols except BGP routes, which get a metric of 1.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to cause BGP routes to be redistributed into an OSPF domain:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 110
RP/0/RP0/CPU0:router(config-ospf)# redistribute bgp 100
```

The following example shows how to redistribute the specified IS-IS process routes into an OSPF domain. The IS-IS routes are redistributed with a metric of 100.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# redistribute isis 108 metric 100
```

In the following example, network 10.0.0.0 appears as an external link-state advertisement (LSA) in OSPF 1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-if)# ip address 10.0.0.0 255.0.0.0

RP/0/RP0/CPU0:router(config)# interface tenGigE 0/2/0/2
RP/0/RP0/CPU0:router(config)# ip address 10.99.0.0 255.0.0.0

RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# redistribute ospf 2
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/2/0/2

RP/0/RP0/CPU0:router(config)# router ospf 2
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/1/0/1
```
retransmit-interval (OSPF)

To specify the time between link-state advertisement (LSA) retransmissions for adjacencies belonging to the Open Shortest Path First (OSPF) interface, use the `retransmit-interval` command in the appropriate mode. To return to the default value, use the `no` form of this command.

```
retransmit-interval  seconds
no  retransmit-interval
```

**Syntax Description**

| `seconds` | Time (in seconds) between retransmissions. It must be greater than the expected round-trip delay between any two routers on the attached network. Range is 1 to 65535 seconds. |

**Command Default**

- If this command is not specified in interface configuration mode, then the interface adopts the retransmit interval parameter specified by the area.
- If this command is not specified in area configuration mode, then the interface adopts the retransmit interval parameter specified for the process.
- If this command is not specified at any level, then the default retransmit interval is 5 seconds.

**Command Modes**

- Interface configuration
- Area configuration
- XR Config mode
- Virtual-link configuration
- Multi-area configuration
- Sham-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

- When a router sends an LSA to its neighbor, it keeps the LSA until it receives the acknowledgment message. If the router receives no acknowledgment, it resends the LSA.
- The setting of this parameter should be conservative, or needless retransmission results. The value should be larger for serial lines and virtual links.

**Examples**

The following example shows how to set the retransmit interval value to 8 seconds in interface configuration mode:

```
``
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-ospf-ar-if)# retransmit-interval 8
route-policy (OSPF)

To specify a routing policy to filter Type 3 link-state advertisements (LSA), use the `route-policy` command in area configuration mode. To disable the routing policy, use the `no` form of this command.

```
route-policy  route-policy-name  {in|out}
no  route-policy  route-policy-name  {in|out}
```

**Syntax Description**
- `route-policy-name` Name of route policy.
- `in` Applies policy to inbound routes.
- `out` Applies policy to outbound routes.

**Command Default**
No policy is applied.

**Command Modes**
Area configuration

**Command History**
Release 6.0 This command was introduced.

**Usage Guidelines**
Use the `route-policy` command to specify an OSPF routing policy for an inbound or outbound route. The policy can be used to filter routes or modify route attributes.

**Task ID**
- `ospf` read, write

**Examples**
The following example shows how to specify an OSPF route policy for inbound routes in area 0:

```
RP/0/RP0/CPU0:router#  configure
RP/0/RP0/CPU0:router(config)#  router ospf 109
RP/0/RP0/CPU0:router(config-ospf)#  area 0
RP/0/RP0/CPU0:router(config-ospf-area)#  route-policy area0_in in
```
router-id (OSPF)

To configure a router ID for the Open Shortest Path First (OSPF) process, use the `router-id` command in the appropriate mode. To cause the software to use the default method of determining the router ID, use the `no` form of this command after clearing or restarting the OSPF process.

```
router-id router-id
no router-id router-id
```

**Syntax Description**

```
router-id  32-bit router ID value specified in four-part, dotted-decimal notation.
```

**Command Default**

If this command is not configured, the router ID is the highest IP version 4 (IPv4) address for an interface on the router, with any loopback interface taking precedence.

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

It is good practice to use the `router-id` command to explicitly specify a unique 32-bit numeric value for the router ID. This action ensures that OSPF can function regardless of the interface address configuration. Clear the OSPF process using the `clear ospf process` command or restart the OSPF process for the `no router-id` command to take effect.

OSPF attempts to obtain a router ID in the following ways (in order of preference):

1. By default, when the OSPF process initializes, it checks if there is a router-id in the checkpointing database.
2. The 32-bit numeric value specified by the OSPF `router-id` command in router configuration mode. (This value can be any 32-bit value. It is not restricted to the IPv4 addresses assigned to interfaces on this router, and need not be a routable IPv4 address.)
3. The ITAL selected router-id.
4. The primary IPv4 address of an interface over which this OSPF process is running. The first interface address in the OSPF interface is selected.

Unlike OSPF version 3, OSPF version 2 is guaranteed to have at least one interface with an IPv4 address configured.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to assign the IP address of 172.20.10.10 to the OSPF process 109:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config-ospf)# router-id 172.20.10.10
```
router ospf

To configure an Open Shortest Path First (OSPF) routing process, use the `router ospf` command in XR Config mode. To terminate an OSPF routing process, use the `no` form of this command.

```bash
router ospf process-name
no router ospf process-name
```

**Syntax Description**

| process-name | Name that uniquely identifies an OSPF routing process. The process name is any alphanumeric string no longer than 40 characters without spaces. |

**Command Default**

No OSPF routing process is defined.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You can specify multiple OSPF routing processes in each router. Up to 10 processes can be configured. The recommendation is not to exceed 4 OSPF processes.

All OSPF configuration commands must be configured under an OSPF routing process. For example, two of these commands are the `default-metric` command and the `router-id` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
<tr>
<td>rib</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to instantiate an OSPF routing process called 109:

```bash
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
```
sham-link

To configure an Open Shortest Path First OSPF sham link between two provider edge routers, use the **sham-link** command in VRF area configuration mode. To terminate an (OSPF) sham link, use the **no** form of this command.

**sham-link source-address destination-address**

**no sham-link**

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>source-address</td>
<td>IP address of the local (source) sham-link endpoint specified in four-part, dotted-decimal notation.</td>
<td></td>
</tr>
<tr>
<td>destination-address</td>
<td>IP address of the remote (destination) sham-link endpoint specified in four-part, dotted-decimal notation.</td>
<td></td>
</tr>
</tbody>
</table>

**Command Default**

No sham link is configured.

**Command Modes**

VRF area configuration.

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **sham-link** command to configure a point-to-point connection between two provider edge (PE) routers creating an interconnect between two VPN sites (VPN backbone). Sham links are configured on PE provider edge (PE) routers in a Multiprotocol Label Switching (MPLS) VPN backbone.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure an OSPF sham link:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 109
RP/0/RP0/CPU0:router(config_ospf)# vrf vrf_a
RP/0/RP0/CPU0:router(config_ospf_vrf)# area 0
RP/0/RP0/CPU0:router(config_ospf_vrf_ar)# sham-link 192.168.40.0 172.16.30.0
RP/0/RP0/CPU0:router(config_ospf_vrf_ar_sl)# cost 23
```
show ospf

To display general information about Open Shortest Path First (OSPF) routing processes, use the `show ospf` command in XR EXEC mode.

```
show ospf [process-name] [summary]
```

**Syntax Description**

- **process-name**  (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only information for the specified routing process is displayed.

**Command Default**

IPv4 and unicast address prefixes

**Command Modes**

- XR EXEC mode
- OSPFv3

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show ospf` command to provide basic information about the OSPF processes running on the router. Additional options provide in-depth information.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

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

```
RP/0/RP0/CPU0:router# show ospf

Routing Process "ospf 1" with ID 1.1.1.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
It is an area border router
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPF's 10000 msecs
Maximum wait time between two consecutive SPF's 10000 msecs
Initial LSA throttle delay 500 msecs
Minimum hold time for LSA throttle 5000 msecs
Maximum wait time for LSA throttle 5000 msecs
Minimum LSA interval 5000 msecs. Minimum LSA arrival 1 secs
Maximum number of configured interfaces 255
Number of external LSA 0. Checksum Sum 00000000
Number of opaque AS LSA 0. Checksum Sum 00000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 2. 2 normal 0 stub 0 nssa
External flood list length 0
```
Non-Stop Forwarding enabled

Area BACKBONE(0) (Inactive)
Number of interfaces in this area is 2
SPF algorithm executed 8 times
Number of LSA 2. Checksum Sum 0x01ba83
Number of opaque link LSA 0. Checksum Sum 00000000
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0

Area 1
Number of interfaces in this area is 1
SPF algorithm executed 9 times
Number of LSA 2. Checksum Sum 0x0153ea
Number of opaque link LSA 0. Checksum Sum 00000000
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0

This table describes the significant fields shown in the display.

Table 33: show ospf Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing Process “ospf 201” with ID 172.22.110.200</td>
<td>OSPF process name.</td>
</tr>
<tr>
<td>Supports only</td>
<td>Number of types of service supported (Type 0 only).</td>
</tr>
<tr>
<td>It is</td>
<td>Types are internal, area border, or autonomous system boundary.</td>
</tr>
<tr>
<td>Redistributing External Routes from</td>
<td>Lists of redistributed routes, by protocol.</td>
</tr>
<tr>
<td>SPF schedule delay</td>
<td>Delay time of SPF calculations.</td>
</tr>
<tr>
<td>Minimum LSA interval</td>
<td>Minimum interval between LSAs.</td>
</tr>
<tr>
<td>Minimum LSA arrival</td>
<td>Minimum elapsed time between accepting an update for the same link-state advertisement (LSA).</td>
</tr>
<tr>
<td>external LSA</td>
<td>Total number of Type 5 LSAs in the LSDB.</td>
</tr>
<tr>
<td>opaque LSA</td>
<td>Total number of Type 10 LSAs in the LSDB.</td>
</tr>
<tr>
<td>DCbitless...AS LSA</td>
<td>Total number of Demand Circuit Type 5 and Type 11 LSAs.</td>
</tr>
<tr>
<td>DoNotAge...AS LSA</td>
<td>Total number of Type 5 and Type 11 LSAs with the DoNotAge bit set.</td>
</tr>
<tr>
<td>Number of areas</td>
<td>Number of areas in router, area addresses, and so on.</td>
</tr>
<tr>
<td>Area BACKBONE</td>
<td>Backbone is area 0.</td>
</tr>
</tbody>
</table>
show ospf border-routers

To display the internal Open Shortest Path First (OSPF) routing table entries to an Area Border Router (ABR) and Autonomous System Boundary Router (ASBR), use the `show ospf border-routers` command in XR EXEC mode.

```
show ospf [process-name] border-routers [router-id]
```

**Syntax Description**

`process-name` (Optional) OSPF process name. If this argument is included, only information for the specified routing process is included.

**Command Default**

IPv4 and unicast address prefixes

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show ospf border-routers` command to list all OSPF border routers visible to the specified processes and to ascertain the OSPF topology of the router.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show ospf border-routers` command:

```
RP/0/RP0/CPU0:router# show ospf border-routers

OSPF 1 Internal Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 172.31.97.53 [1] via 172.16.1.53, GigabitEthernet 3/0/0/0, ABR/ASBR , Area 0, SPF 3
```

This table describes the significant fields shown in the display.

**Table 34: show ospf border-routers Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>Type of this route; i indicates an intra-area route, I an interarea route.</td>
</tr>
<tr>
<td>172.31.97.53</td>
<td>Router ID of destination.</td>
</tr>
<tr>
<td>[1]</td>
<td>Cost of using this route.</td>
</tr>
</tbody>
</table>
### show ospf border-routers

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>172.16.1.53</td>
<td>Next-Next hop toward the destination.</td>
</tr>
<tr>
<td>GigabitEthernet 3/0/0/0</td>
<td>Packets destined for 172.16.1.53 are sent over GigabitEthernet interface 3/0/0/0.</td>
</tr>
<tr>
<td>ABR/ASBR</td>
<td>Router type of the destination; it is either an Area Border Router (ABR) or Autonomous System Boundary Router (ASBR) or both.</td>
</tr>
<tr>
<td>Area 0</td>
<td>Area ID of the area from which this route is learned.</td>
</tr>
<tr>
<td>SPF 3</td>
<td>Internal number of the shortest path first (SPF) calculation that installs this route.</td>
</tr>
</tbody>
</table>
show ospf database

To display lists of information related to the Open Shortest Path First (OSPF) database for a specific router, use the `show ospf database` command in XR EXEC mode.

Note

VRF is not supported.

```
show ospf [process-name] [vrf {vrf-name}all] [area-id] database
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [adv-router ip-address]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [asbr-summary] [link-state-id]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [asbr-summary] [link-state-id] [internal] [adv-router {ip-address}]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [external] [link-state-id]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [external] [link-state-id] [internal] [adv-router {ip-address}]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [network] [link-state-id]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [network] [link-state-id] [internal] [adv-router {ip-address}]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [nssa-external] [link-state-id]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [nssa-external] [link-state-id] [internal] [adv-router {ip-address}]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [opaque-area] [link-state-id]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [opaque-area] [link-state-id] [internal] [adv-router {ip-address}]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [opaque-as] [link-state-id]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [opaque-as] [link-state-id] [internal] [adv-router {ip-address}]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [opaque-link] [link-state-id]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [opaque-link] [link-state-id] [internal] [adv-router {ip-address}]
show ospf [process-name] [vrf {vrf-name}all] [area-id] database [router] [link-state-id]
```
show ospf [process-name] [vrf {vrf-name|all}] [area-id] database [router] [internal] [adv-router [ip-address]]

Syntax Description

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>process-name</td>
<td>(Optional) OSPF process name that uniquely identifies an OSPF routing process. The process name is any alphanumeric string no longer than 40 characters. If this argument is included, only information for the specified routing process is included.</td>
</tr>
<tr>
<td>vrf</td>
<td>(Optional) Specifies an OSPF VPN routing and forwarding (VRF) instance.</td>
</tr>
<tr>
<td>vrf-name</td>
<td>(Optional) Name of the OSPF VRF. The vrf-name argument can be specified as an arbitrary string. The strings “default” and “all” are reserved VRF names.</td>
</tr>
<tr>
<td>all</td>
<td>(Optional) Specifies all OSPF VRF instances.</td>
</tr>
<tr>
<td>area-id</td>
<td>(Optional) Area number used to define the particular area.</td>
</tr>
<tr>
<td>adv-router ip-address</td>
<td>(Optional) Displays all LSAs of the specified router.</td>
</tr>
<tr>
<td>asbr-summary</td>
<td>(Optional) Displays information only about the Autonomous System Boundary Router (ASBR) summary LSAs.</td>
</tr>
<tr>
<td>link-state-id</td>
<td>(Optional) Portion of the Internet environment that is being described by the advertisement. The value entered depends on the link-state type of the advertisement. It must be entered in the form of an IP address.</td>
</tr>
</tbody>
</table>

When the link-state advertisement (LSA) is describing a network, the link-state-id can take one of two forms:

- The network IP address (as in Type 3 summary link advertisements and in autonomous system external link advertisements).
- A derived address obtained from the link-state ID.

Note
Masking the link-state ID of a network link advertisement with the subnet mask of the network yields the IP address of the network.

When the LSA is describing a router, the link-state ID is always the OSPF router ID of the described router.

When an autonomous system external advertisement (LS Type = 5) is describing a default route, its link-state ID is set to Default Destination (0.0.0.0).

internal | (Optional) Displays internal LSA information. |

self-originate | (Optional) Displays only self-originated LSAs (from the local router). |

database-summary | (Optional) Displays how many of each type of LSA for each area there are in the database and the total. |
### Command Default
IPv4 and unicast address prefixes

### Command Modes
XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| Release 6.0 | This command was introduced.

### Usage Guidelines
The various forms of the `show ospf database` command deliver information about different OSPF link-state advertisements. This command can be used to examine the link-state database (LSD) and its contents. Each router participating in an area having identical database entries pertaining to that area (with the exception of LSAs that are being flooded). Numerous options (such as `network` and `router`) are used to display portions of the database.

### Examples
The following is sample output from the `show ospf database` command when no arguments or keywords are used:

```
RP/0/RP0/CPU0:router# show ospf database

OSPF Router with ID (172.20.1.11) (Process ID 1)

Router Link States (Area 0)

<table>
<thead>
<tr>
<th>Link ID</th>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Checksum Link count</th>
</tr>
</thead>
<tbody>
<tr>
<td>172.20.1.8</td>
<td>172.20.1.8</td>
<td>1381</td>
<td>0x8000010D</td>
<td>0xEF60</td>
</tr>
</tbody>
</table>
```
Net Link States (Area 0)

<table>
<thead>
<tr>
<th>Link ID</th>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>172.22.1.27</td>
<td>172.20.1.27</td>
<td>1323</td>
<td>0x8000005B</td>
<td>0x12CC</td>
</tr>
<tr>
<td>172.20.1.27</td>
<td>172.20.1.11</td>
<td>1460</td>
<td>0x8000002FE</td>
<td>0xEB3D</td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

**Table 35: show ospf database Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Link ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>ADV Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>Age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Seq#</td>
<td>Link-state sequence number (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Fletcher checksum of the complete contents of the LSA.</td>
</tr>
<tr>
<td>Link count</td>
<td>Number of interfaces detected for the router.</td>
</tr>
<tr>
<td>Opaque ID</td>
<td>Opaque LSA ID number.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospf database` command with the `asbr-summary` keyword:

```
RP/0/RSPORP0/CPU0:router# show ospf database asbr-summary
OSPF Router with ID (192.168.0.1) (Process ID 300)
Summary ASB Link States (Area 0.0.0.0)
  LS age: 1463
  Options: (No TOS-capability)
  LS Type: Summary Links (AS Boundary Router)
  Link State ID: 172.17.245.1 (AS Boundary Router address)
  Advertising Router: 172.17.241.5
  LS Seq Number: 80000072
  Checksum: 0x3548
  Length: 28
  Network Mask: /0
  TOS: 0 Metric: 1
```
This table describes the significant fields shown in the display.

**Table 36: show ospf database asbr-summary Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPF process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only).</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID (ASBR).</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSAs.</td>
</tr>
<tr>
<td>Network Mask</td>
<td>Network mask implemented.</td>
</tr>
<tr>
<td>TOS</td>
<td>Type of service.</td>
</tr>
<tr>
<td>Metric</td>
<td>Link-state metric.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospf database` command with the `external` keyword:

```plaintext
RP/0/RP0/CPU0:router# show ospf database external

OSPF Router with ID (192.168.0.1) (Process ID 300)

    Type-5 AS External Link States

    LS age: 280
    Options: (No TOS-capability)
    LS Type: AS External Link
    Link State ID: 172.17.0.0 (External Network Number)
    Advertising Router: 172.17.70.6
    LS Seq Number: 80000AFD
    Checksum: 0xC3A
    Length: 36
    Network Mask: 255.255.0.0
        Metric Type: 2 (Larger than any link state path)
        TOS: 0
        Metric: 1
        Forward Address: 0.0.0.0
        External Route Tag: 0
```
This table describes the significant fields shown in the display.

**Table 37: show ospf database external Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF Router with Router ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPF process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only).</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID (external network number).</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence number (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Network Mask</td>
<td>Network mask implemented.</td>
</tr>
<tr>
<td>Metric Type</td>
<td>External type.</td>
</tr>
<tr>
<td>TOS</td>
<td>Type of service.</td>
</tr>
<tr>
<td>Metric</td>
<td>Link-state metric.</td>
</tr>
<tr>
<td>Forward Address</td>
<td>Forwarding address. Data traffic for the advertised destination is forwarded</td>
</tr>
<tr>
<td></td>
<td>to this address. If the forwarding address is set to 0.0.0.0, data traffic is</td>
</tr>
<tr>
<td></td>
<td>forwarded instead to the originator of the advertisement.</td>
</tr>
<tr>
<td>External Route Tag</td>
<td>External route tag, a 32-bit field attached to each external route. This tag</td>
</tr>
<tr>
<td></td>
<td>is not used by the OSPF protocol itself.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospf database` command with the `network` keyword:

```
RP/0/RP0/CPU0:router# show ospf database network

OSPF Router with ID (192.168.0.1) (Process ID 300)
Net Link States (Area 0.0.0.0)
  LS age: 1367
  Options: (No TOS-capability)
  LS Type: Network Links
  Link State ID: 172.23.1.3 (address of Designated Router)
  Advertising Router: 192.168.0.1
```
LS Seq Number: 800000E7  
Checksum: 0x1229  
Length: 52  
Network Mask: /24  
  Attached Router: 192.168.0.1  
  Attached Router: 172.23.241.5  
  Attached Router: 172.23.1.1  
  Attached Router: 172.23.54.5  
  Attached Router: 172.23.1.5

This table describes the significant fields shown in the display.

**Table 38: show ospf database network Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPF process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only).</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID of the designated router.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence number (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Network Mask</td>
<td>Network mask implemented.</td>
</tr>
<tr>
<td>Attached Router</td>
<td>List of routers attached to the network, by IP address.</td>
</tr>
</tbody>
</table>

The following is sample output, carrying Multiprotocol Label Switching traffic engineering (MPLS TE) specification information, from the `show ospf database` command with the `opaque-area` keyword and a `link-state-id` of adv-router:

```
RP/0/RP0/CPU0:router# show ospf database opaque-area adv-router 172.20.1.12

OSPF Router with ID (172.20.1.11) (Process ID 1)
  Type-10 Opaque Link Area Link States (Area 0)
    LS age: 224
    Options: (No TOS-capability, DC)
    LS Type: Opaque Area Link
    Link State ID: 1.0.0.0
    Opaque Type: 1
    Opaque ID: 0
    Advertising Router: 172.20.1.12
```
The following is the sample output from the `show ospf database opaque-area` command displaying the extended link LSA information.

```
RP/0/RP0/CPU0:router# show ospf database opaque-area 4.0.0.0
LS age: 361
Options: (No TOS-capability, DC)
LS Type: Opaque Area Link
Link State ID: 8.0.0.40
Opaque Type: 8
Opaque ID: 40
Advertising Router: 100.0.0.3
LS Seq Number: 8000012e
Checksum: 0xeab4
Length: 92

Extended Link TLV: Length: 68
  Link-type : 2
  Link ID : 100.0.9.4
  Link Data : 100.0.9.3

LAN Adj sub-TLV: Length: 16
  Flags : 0x0
  MTID : 0
  Weight : 0
  Neighbor ID: 100.0.0.1

  SID/Label sub-TLV: Length: 3
    SID : 24001

LAN Adj sub-TLV: Length: 16
  Flags : 0x0
  MTID : 0
  Weight : 0
  Neighbor ID: 100.0.0.2

  SID/Label sub-TLV: Length: 3
    SID : 24000
```
The following is sample output from the `show ospf database` command that displays a Type 10, Router Information LSA:

```
RP/0/RP0/CPU0:router# show ospf database opaque-area 4.0.0.0

OSPF Router with ID (3.3.3.3) (Process ID orange)

Type-10 Opaque Link Area Link States (Area 0)

LS age: 105
Options: (No TOS-capability, DC)
LS Type: Opaque Area Link
Link State ID: 4.0.0.0
Opaque Type: 4
Opaque ID: 0
Advertising Router: 3.3.3.3
LS Seq Number: 80000052
Checksum: 0x34e2
Length: 52
Fragment number: 0

Router Information TLV: Length: 4
Capabilities:
  Graceful Restart Helper Capable
  Traffic Engineering enabled area
  All capability bits: 0x50000000

PCE Discovery TLV: Length: 20
IPv4 Address: 3.3.3.3
PCE Scope: 0x20000000
Compute Capabilities:
  Inter-area default (Rd-bit)
Compute Preferences:
  Intra-area: 0 Inter-area: 0
  Inter-AS: 0 Inter-layer: 0
```

This table describes the significant fields shown in the display.

**Table 39: show ospf database opaque-area Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPF process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only).</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID.</td>
</tr>
<tr>
<td>Opaque Type</td>
<td>Opaque link-state type.</td>
</tr>
<tr>
<td>Opaque ID</td>
<td>Opaque ID number.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Fragment number</td>
<td>Arbitrary value used to maintain multiple traffic engineering LSAs.</td>
</tr>
<tr>
<td>Link ID</td>
<td>Link ID number.</td>
</tr>
<tr>
<td>Interface Address</td>
<td>ID address of the interface.</td>
</tr>
<tr>
<td>Neighbor Address</td>
<td>IP address of the neighbor.</td>
</tr>
<tr>
<td>Admin Metric</td>
<td>Administrative metric value used by MPLS TE.</td>
</tr>
<tr>
<td>Maximum bandwidth</td>
<td>Specifies maximum bandwidth (in kbps).</td>
</tr>
<tr>
<td>Maximum reservable bandwidth</td>
<td>Specifies maximum reservable bandwidth (in kbps).</td>
</tr>
<tr>
<td>Number of Priority</td>
<td>Priority number.</td>
</tr>
<tr>
<td>Affinity Bit</td>
<td>Used by MPLS TE.</td>
</tr>
<tr>
<td>Router Information TLV</td>
<td>Router capabilities are advertised in this TLV.</td>
</tr>
<tr>
<td>Capabilities</td>
<td>Some router capabilities include stub router, traffic engineering, graceful</td>
</tr>
<tr>
<td></td>
<td>restart, and graceful restart helper.</td>
</tr>
<tr>
<td>PCE Discovery TLV</td>
<td>PCE address and capability information is advertised in this TLV.</td>
</tr>
<tr>
<td>IPv4 Address</td>
<td>Configured PCE IPv4 address.</td>
</tr>
<tr>
<td>PCE Scope</td>
<td>Computation capabilities of the PCE.</td>
</tr>
<tr>
<td>Compute Capabilities</td>
<td>Compute capabilities and preferences of the PCE.</td>
</tr>
<tr>
<td>Inter-area default (RD-bit)</td>
<td>PCE compute capabilities such as intra-area, inter-area, inter-area default,</td>
</tr>
<tr>
<td></td>
<td>inter-AS, inter-AS default and inter-layer.</td>
</tr>
<tr>
<td>Compute Preferences</td>
<td>Order or preference of path computation that includes intra-area, inter-area,</td>
</tr>
<tr>
<td></td>
<td>inter-AS, and inter-layer preferences.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospf database` command with the `router` keyword:
show ospf database router

OSPF Router with ID (192.168.0.1) (Process ID 300)

Router Link States (Area 0.0.0.0)

LS age: 1176
Options: (No TOS-capability)
LS Type: Router Links
Link State ID: 172.23.21.6
Advertising Router: 172.23.21.6
LS Seq Number: 80002CF6
Checksum: 0x73B7
Length: 120
AS Boundary Router
Number of Links: 8

Link connected to: another Router (point-to-point)
(Link ID) Neighboring Router ID: 172.23.21.5
(Link Data) Router Interface address: 172.23.21.6
Number of TOS metrics: 0
TOS 0 Metrics: 2

This table describes the significant fields shown in the display.

Table 40: show ospf database router Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPF process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only).</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>AS Boundary Router</td>
<td>Definition of router type.</td>
</tr>
<tr>
<td>Number of Links</td>
<td>Number of active links.</td>
</tr>
<tr>
<td>Link ID</td>
<td>Link type.</td>
</tr>
<tr>
<td>Link Data</td>
<td>Router interface address.</td>
</tr>
</tbody>
</table>
The following is sample output from `show ospf database` command with the `summary` keyword:

```
RP/0/RP0/CPU0:router# show ospf database summary

OSPF Router with ID (192.168.0.1) (Process ID 300)

Summary Net Link States (Area 0.0.0.0)
  LS age: 1401
  Options: (No TOS-capability)
  LS Type: Summary Links (Network)
  Link State ID: 172.23.240.0 (Summary Network Number)
  Advertising Router: 172.23.241.5
  LS Seq Number: 80000072
  Checksum: 0x84FF
  Length: 28
  Network Mask: /24
  TOS: 0  Metric: 1
```

This table describes the significant fields shown in the display.

### Table 41: `show ospf database summary` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPF Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPF process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only).</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID (summary network number).</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Network Mask</td>
<td>Network mask implemented.</td>
</tr>
<tr>
<td>TOS</td>
<td>Type of service.</td>
</tr>
<tr>
<td>Metric</td>
<td>Link-state metric.</td>
</tr>
</tbody>
</table>
The following is sample output from `show ospf database` command with the `database-summary` keyword:

```
RP/0/RP0/CPU0:router# show ospf database database-summary

OSPF Router with ID (172.19.65.21) (Process ID 1)

Area 0 database summary
  LSA Type    Count  Delete  Maxage
  Router      2       0       0
  Network     1       0       0
  Summary Net 2       0       0
  Summary ASBR 0       0       0
  Type-7 Ext  0       0       0
  Opaque Link 0       0       0
  Opaque Area 0       0       0
  Subtotal    5       0       0

Process 1 database summary
  LSA Type    Count  Delete  Maxage
  Router      2       0       0
  Network     1       0       0
  Summary Net 2       0       0
  Summary ASBR 0       0       0
  Type-7 Ext  0       0       0
  Opaque Link 0       0       0
  Opaque Area 0       0       0
  Type-5 Ext  2       0       0
  Opaque AS   0       0       0
  Total       7       0       0
```

This table describes the significant fields shown in the display.

**Table 42: show ospf database database-summary Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSA Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Count</td>
<td>Number of advertisements in that area for each link-state type.</td>
</tr>
<tr>
<td>Delete</td>
<td>Number of LSAs that are marked “Deleted” in that area.</td>
</tr>
<tr>
<td>Maxage</td>
<td>Number of LSAs that are marked “Maxaged” in that area.</td>
</tr>
</tbody>
</table>
show ospf flood-list

To display a list of Open Shortest Path First (OSPF) link-state advertisements (LSAs) waiting to be flooded over an interface, use the **show ospf flood-list** command in XR EXEC mode.

**show ospf [process-name] [area-id] flood-list [type interface-path-id]**

**Syntax Description**

- **process-name** (Optional) OSPF process name that uniquely identifies an OSPF routing process. The process name is any alphanumeric string no longer than 40 characters. If this argument is included, only information for the specified routing process is included.
- **area-id** (Optional) Area number used to define the particular area.
- **type** Interface type. For more information, use the question mark (?) online help function.
- **interface-path-id** Physical interface or virtual interface.

**Note**
Use the **show interfaces** command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

**Command Default**
All interfaces

**Command Modes**
XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the **show ospf flood-list** command to display LSAs in flood queue and queue length.

Flood list information is transient and normally the flood lists are empty.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the **show ospf flood-list** command for interface GigabitEthernet 3/0/0/0:

```
RP/0/RP0/CPU0:router# show ospf flood-list tenGigE 3/0/0/0

Interface GigabitEthernet3/0/0/0, Queue length 20
Link state retransmission due in 12 msec
Displaying 6 entries from flood list:
```
This table describes the significant fields shown in the display.

**Table 43: show ospf flood-list Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GigabitEthernet3/0/0/0</td>
<td>Interface for which information is displayed.</td>
</tr>
<tr>
<td>Queue length</td>
<td>Number of LSAs waiting to be flooded.</td>
</tr>
<tr>
<td>Link state retransmission due in</td>
<td>Length of time (in milliseconds) before next link-state transmission.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of LSA.</td>
</tr>
<tr>
<td>LS ID</td>
<td>Link-state ID of the LSA.</td>
</tr>
<tr>
<td>ADV RTR</td>
<td>IP address of the advertising router.</td>
</tr>
<tr>
<td>Seq NO</td>
<td>Sequence number of the LSA.</td>
</tr>
<tr>
<td>Age</td>
<td>Age of the LSA (in seconds).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Checksum of the LSA.</td>
</tr>
</tbody>
</table>
show ospf interface

To display strict-mode information use the `show ospf interface` command in XR EXEC mode.

```
show ospf interface [interface  type interface-path-id]
```

**Syntax Description**

- `type` Interface type. For more information, use the question mark (?) online help function.
- `interface-path-id` Physical interface or virtual interface.

**Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

**Command Default**

XR EXEC mode

**Command Modes**

- EXEC mode

**Command History**

- **Release 6.0**  This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgp</td>
<td>read</td>
</tr>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**show ospf interface**

The following example shows the output from the `show ospf interface` command.

```
RP/0/0/CPU0:rt2(config-ospf-ar-if)# show ospf interface tenGigE 0/2/0/0
Sun Feb 15 12:17:35.072 IST
tenGigE 0/2/0/0 is up, line protocol is up
  Internet Address 10.1.1.2/24, Area 0
  Process ID 1, Router ID 2.2.2.2, Network Type BROADCAST, Cost: 1
  Transmit Delay is 1 sec, State DR, Priority 1, MTU 1500, MaxPktSz 1500
  BFD enabled, BFD interval 150 maec, BFD multiplier 3, Mode: Strict
  Designated Router (ID) 2.2.2.2, Interface address 10.1.1.2
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Hello due in 00:00:07:358
  Index 1/1, flood queue length 0
  Next 0(0)/0(0)
  Last flood scan length is 1, maximum is 1
  Last flood scan time is 0 msec, maximum is 0 maec
```
LS Ack List: current length 0, high water mark 1
Neighbor Count is 1, Adjacent neighbor count is 0
Suppress hello for 0 neighbor(s)
Multi-area interface Count is 0
show ospf message-queue

To display the information about the queue dispatch values, peak lengths, and limits, use the show ospf message-queue command in XR EXEC mode.

show ospf message-queue

This command has no arguments or keywords.

Command Default
No default behavior or values

Command Modes
XR EXEC mode

Command History
Release 6.0 This command was introduced.

Usage Guidelines
No specific guidelines impact the use of this command.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the show ospf message-queue command:

```
RP/0/RP0/CPU0:router# show ospf 1 message-queue

OSPF 1
   Hello Input Queue:
       Current queue length: 0
       Event scheduled: 0
       Total queuing failures: 0
       Maximum length : 102
       Pkts pending processing: 0
       Limit: 5000

Router Message Queue
   Current instance queue length: 0
   Current redistribution queue length: 0
   Current ex spf queue length: 0
   Current sum spf queue length: 0
   Current intra spf queue length: 0
   Event scheduled: 0
   Maximum length : 101
   Total low queuing failures: 0
   Total medium queuing failures: 0
   Total high queuing failures: 0
   Total instance events: 919
   Processing quantum : 300
   Low queuing limit: 8000
   Medium queuing limit: 9000
   High queuing limit: 9500
```
Rate-limited LSA processing quantum: 150
Current rate-limited LSA queue length: 0
Rate-limited LSA queue peak len: 517

Rate-limited LSAs processed: 4464
Flush LSA processing quantum: 150
Current flush LSA queue length: 0
Flush LSA queue peak len: 274
Rate-limited flush LSAs processed: 420

SPF-LSA-limit processing quantum: 150
Managed timers processing quantum: 50
Instance message count: 0
Instance pulse send count: 919
Instance pulse received count: 919
Global pulse count: 0
Instance Pulse errors: 0

TE Message Queue
Current queue length: 0
Total queuing failures: 0
Maximum length : 0

Number of Dlink errors: 0

This table describes the significant fields shown in the display.

Table 44: show ospf message-queue Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hello Input Queue</td>
<td>This section provides statistics on the number of events and incoming packets processed in the Hello (incoming packet) thread of the OSPF process.</td>
</tr>
<tr>
<td>Router Message Queue</td>
<td>This section provides statistics on the events and messages processed in the Router (primary) thread of the OSPF process.</td>
</tr>
<tr>
<td>TE Message Queue</td>
<td>This section provides statistics on traffic-engineering events and messages received by OSPF from TE (the te_control process). These events are processed in the Router thread of the OSPF process.</td>
</tr>
<tr>
<td>Number of Dlink errors</td>
<td>The number of enqueuing or dequeuing errors seen across all the linked-lists in the OSPF process.</td>
</tr>
</tbody>
</table>
show ospf neighbor

To display Open Shortest Path First (OSPF) neighbor information on an individual interface basis, use the `show ospf neighbor` command in XR EXEC mode.

```
show ospf [process-name] [area-id] neighbor [type interface-path-id] [neighbor-id] [detail|area-sorted]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>process-name</code></td>
<td>(Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the <code>router ospf</code> command. If this argument is included, only information for the specified routing process is displayed.</td>
</tr>
<tr>
<td><code>area-id</code></td>
<td>(Optional) Area ID. If you do not specify an area, all areas are displayed.</td>
</tr>
<tr>
<td><code>type</code></td>
<td>Interface type. For more information, use the question mark (?) online help function.</td>
</tr>
<tr>
<td><code>interface-path-id</code></td>
<td>Physical interface or virtual interface. Use the <code>show interfaces</code> command to see a list of all interfaces currently configured on the router.</td>
</tr>
<tr>
<td><code>neighbor-id</code></td>
<td>(Optional) Neighbor ID.</td>
</tr>
<tr>
<td><code>detail</code></td>
<td>(Optional) Displays all neighbors given in detail (lists all neighbors).</td>
</tr>
<tr>
<td><code>area-sorted</code></td>
<td>(Optional) Specifies that all neighbors are grouped by area.</td>
</tr>
</tbody>
</table>

### Command Default

All neighbors

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

No specific guidelines impact the use of this command.

### Examples

The following is sample output from the `show ospf neighbor` command showing two lines of summary information for each neighbor:
**show ospf neighbor**

Neighbors for OSPF

<table>
<thead>
<tr>
<th>Neighbor ID</th>
<th>Pri</th>
<th>State</th>
<th>Dead Time</th>
<th>Address</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.199.137</td>
<td>1</td>
<td>FULL/DR</td>
<td>0:00:31</td>
<td>172.31.80.37</td>
<td>tenGigE 0/3/0/2</td>
</tr>
<tr>
<td>Neighbor is up for 18:45:22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.48.1</td>
<td>1</td>
<td>FULL/DROTHER</td>
<td>0:00:33</td>
<td>192.168.48.1</td>
<td>tenGigE 0/3/0/3</td>
</tr>
<tr>
<td>Neighbor is up for 18:45:30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.48.200</td>
<td>1</td>
<td>FULL/DROTHER</td>
<td>0:00:33</td>
<td>192.168.48.200</td>
<td>tenGigE 0/3/0/3</td>
</tr>
<tr>
<td>Neighbor is up for 18:45:25</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.199.137</td>
<td>5</td>
<td>FULL/DR</td>
<td>0:00:33</td>
<td>192.168.48.189</td>
<td>tenGigE 0/3/0/3</td>
</tr>
<tr>
<td>Neighbor is up for 18:45:27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

**Table 45: show ospf neighbor Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor ID</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>Pri</td>
<td>Designated router priority.</td>
</tr>
<tr>
<td>State</td>
<td>OSPF state.</td>
</tr>
<tr>
<td>Dead time</td>
<td>Time (in hh:mm:ss) that must elapse before OSPF declares the neighbor dead.</td>
</tr>
<tr>
<td>Address</td>
<td>Address of next hop.</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface name of next hop.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Amount of time (in hh:mm:ss) that the OSPF neighbor has been up.</td>
</tr>
</tbody>
</table>

The following is sample output showing summary information about the neighbor that matches the neighbor ID:

```
RP/0/RP0/CPU0:router# show ospf neighbor 192.168.199.137

Neighbor 192.168.199.137, interface address 172.31.80.37
  In the area 0.0.0.0 via interface tenGigE 0/3/0/2
  Neighbor priority is 1, State is FULL, 6 state changes
  DR is 0.0.0.0 BDR is 0.0.0.0
  Options is 0x2
  Dead timer due in 0:00:32
  Neighbor is up for 18:45:30
  Number of DBD retrans during last exhange 0
  Index 1/1, retransmission queue length 0, number of retransmission 0
  First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
  Last retransmission scan length is 0, maximum is 0
  Last retransmission scan time is 0 msec, maximum 0 msec
```

```
Neighbor 192.168.199.137, interface address 192.168.48.189
  In the area 0.0.0.0 via interface tenGigE 0/3/0/3
  Neighbor priority is 5, State is FULL, 6 state changes
  Options is 0x2
  Dead timer due in 0:00:32
```
Neighbor is up for 18:45:30
Number of DBD retrans during last exchange 0
Index 1/1, retransmission queue length 0, number of retransmission 0
First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
Last retransmission scan length is 0, maximum is 0
Last retransmission scan time is 0 msec, maximum 0 msec

Total neighbor count: 2

This table describes the significant fields shown in the display.

**Table 46: show ospf neighbor 192.168.199.137 Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>IP address of the interface</td>
<td>IP address of the interface.</td>
</tr>
<tr>
<td>In the area</td>
<td>Area and interface through which the OSPF neighbor is known.</td>
</tr>
<tr>
<td>Neighbor priority</td>
<td>Router priority of neighbor and neighbor state.</td>
</tr>
<tr>
<td>State</td>
<td>OSPF state.</td>
</tr>
<tr>
<td>state changes</td>
<td>Number of state changes for this neighbor.</td>
</tr>
<tr>
<td>DR is</td>
<td>Neighbor ID of the designated router.</td>
</tr>
<tr>
<td>BDR is</td>
<td>Neighbor ID of the backup designated router.</td>
</tr>
<tr>
<td>Options</td>
<td>Hello packet options field contents(E-bit only; possible values are 0 and 2; 2 indicates area is not a stub; 0 indicates area is a stub.)</td>
</tr>
<tr>
<td>Dead timer</td>
<td>Time (in hh:mm:ss) to elapse before OSPF declares the neighbor dead.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Amount of time (in hh:mm:ss) that the OSPF neighbor has been up.</td>
</tr>
<tr>
<td>Number of DBD retrans</td>
<td>Number of re-sent database description packets.</td>
</tr>
<tr>
<td>Index</td>
<td>Index and the remaining lines of this command give detailed information about flooding information received from the neighbor.</td>
</tr>
</tbody>
</table>

If you specify the interface along with the neighbor ID, the software displays the neighbors that match the neighbor ID on the interface, as in the following sample display:

```
RP/0/RP0/CPU0:router# show ospf neighbor tenGigE 0/3/0/2 192.168.199.137

Neighbor 192.168.199.137, interface address 172.31.80.37
In the area 0.0.0.0 via interface tenGigE 0/3/0/2
Neighbor priority is 1, State is FULL, 6 state changes
DR is 0.0.0.0 BDR is 0.0.0.0
Options is 0x2
Dead timer due in 0:00:32
Neighbor is up for 18:45:30
Number of DBD retrans during last exchange 0
Index 1/1, retransmission queue length 0, number of retransmission 0
```
This table describes the significant fields shown in the display.

### Table 47: `show ospf neighbor tenGigE 0/3/0/2 192.168.199.137` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>interface address</td>
<td>IP address of the interface.</td>
</tr>
<tr>
<td>In the area</td>
<td>Area and interface through which the OSPF neighbor is known.</td>
</tr>
<tr>
<td>Neighbor priority</td>
<td>Router priority of the neighbor.</td>
</tr>
<tr>
<td>State</td>
<td>OSPF state.</td>
</tr>
<tr>
<td>state changes</td>
<td>Number of state changes for this neighbor.</td>
</tr>
<tr>
<td>DR is</td>
<td>Neighbor ID of the designated router.</td>
</tr>
<tr>
<td>BDR is</td>
<td>Neighbor ID of the backup designated router.</td>
</tr>
<tr>
<td>Options</td>
<td>Hello packet options field contents (E-bit only; possible values are 0 and 2; 2 indicates area is not a stub; 0 indicates area is a stub)</td>
</tr>
<tr>
<td>Dead timer</td>
<td>Time (in hh:mm:ss) to elapse before OSPF declares the neighbor dead.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Amount of time (in hh:mm:ss) that the OSPF neighbor has been up.</td>
</tr>
<tr>
<td>Number of DBD retrans</td>
<td>Number of re-sent database description packets.</td>
</tr>
<tr>
<td>Index</td>
<td>Index and the remaining lines of this command give detailed information about flooding information received from the neighbor.</td>
</tr>
</tbody>
</table>

You can also specify the interface without the neighbor ID to show all neighbors on the specified interface, as in the following sample display:

```
RP/0/RP0/CPU0:router# show ospf neighbor tenGigE POS 0/3/0/3
Neighbors for OSPF ospf1

<table>
<thead>
<tr>
<th>ID</th>
<th>Pri</th>
<th>State</th>
<th>Dead Time</th>
<th>Address</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.48.1</td>
<td>1</td>
<td>FULL/DROTHER</td>
<td>00:00:33</td>
<td>192.168.48.1</td>
<td>tenGigE POS 0/3/0/3</td>
</tr>
<tr>
<td>Neighbor is up for 10:00:33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.48.200</td>
<td>1</td>
<td>FULL/DROTHER</td>
<td>00:03:22</td>
<td>192.168.48.200</td>
<td>tenGigE POS 0/3/0/3</td>
</tr>
<tr>
<td>Neighbor is up for 10:03:22</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.199.137</td>
<td>5</td>
<td>FULL/DR</td>
<td>00:00:32</td>
<td>192.168.48.139</td>
<td>tenGigE POS 0/3/0/3</td>
</tr>
<tr>
<td>Neighbor is up for 10:00:32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Total neighbor count: 3
```
This table describes the significant fields shown in the display.

Table 48: show ospf neighbor tenGigE 0/3/0/3 Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>Pri</td>
<td>Route priority of the neighbor.</td>
</tr>
<tr>
<td>State</td>
<td>OSPF state.</td>
</tr>
<tr>
<td>Dead Time</td>
<td>Time (in hh:mm:ss) to elapse before OSPF declares the neighbor dead.</td>
</tr>
<tr>
<td>Address</td>
<td>Address of next hop.</td>
</tr>
<tr>
<td>Interface</td>
<td>Interface name of next hop.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Time (in hh:mm:ss) that the OSPF neighbor has been up.</td>
</tr>
<tr>
<td>Options</td>
<td>Hello packet options field contents (E-bit only; possible values are 0 and 2; 2 indicates area is not a stub; 0 indicates area is a stub)</td>
</tr>
<tr>
<td>Dead timer</td>
<td>Time (in hh:mm:ss) to elapse before OSPF declares the neighbor dead.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Amount of time (in hh:mm:ss) that the OSPF neighbor has been up.</td>
</tr>
<tr>
<td>Number of DBD retrans</td>
<td>Number of re-sent database description packets.</td>
</tr>
<tr>
<td>Index</td>
<td>Index and the remaining lines of this command give detailed information about flooding information received from the neighbor.</td>
</tr>
</tbody>
</table>

The following samples are from output from the `show ospf neighbor detail` command:

```
RP/0/RP0/CPU0:router# show ospf neighbor detail
Neighbor 192.168.199.137, interface address 172.31.80.37
  In the area 0.0.0.0 via interface tenGigE 0/3/0/2
  Neighbor priority is 1, State is FULL, 6 state changes
  DR is 0.0.0.0 BDR is 0.0.0.0
  Options is 0x2
  Dead timer due in 0:00:32
  Neighbor is up for 18:45:30
  Number of DBD retrans during last exhange 0
  Index 1/1, retransmission queue length 0, number of retransmission 0
  First 0x0(0)/0x0(0) Next 0x0(0)/0x0(0)
  Last retransmission scan length is 0, maximum is 0
  Last retransmission scan time is 0 msec, maximum 0 msec

Total neighbor count: 1

Neighbor 10.1.1.1, interface address 192.168.13.1
  In the area 0 via interface tenGigE 0/3/0/1
  Neighbor priority is 1, State is FULL, 10 state changes
  DR is 0.0.0.0 BDR is 0.0.0.0
```
Options is 0x52
LLS Options is 0x1 (LR)
Dead timer due in 00:00:36
Neighbor is up for 1w2d
Number of DBD retrans during last exchange 0
Index 3/3, retransmission queue length 0, number of retransmission 5
First 0(0)/0(0) Next 0(0)/0(0)
Last retransmission scan length is 1, maximum is 1
Last retransmission scan time is 0 msec, maximum is 0 msec

Neighbor 10.4.4.4, interface address 192.168.34.4
In the area 0 via interface tenGigE 0/3/0/2
Neighbor priority is 1, State is FULL, 48 state changes
DR is 0.0.0.0 BDR is 0.0.0.0
Options is 0x12
LLS Options is 0x1 (LR)
Dead timer due in 00:00:30
Neighbor is up for 00:40:03
Number of DBD retrans during last exchange 0
Index 2/2, retransmission queue length 0, number of retransmission 6
First 0(0)/0(0) Next 0(0)/0(0)
Last retransmission scan length is 0, maximum is 1
Last retransmission scan time is 0 msec, maximum is 0 msec

This table describes the significant fields shown in the display.

**Table 49: show ospf neighbor detail Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>interface address</td>
<td>IP address of the interface.</td>
</tr>
<tr>
<td>In the area</td>
<td>Area and interface through which the OSPF neighbor is known.</td>
</tr>
<tr>
<td>Neighbor priority</td>
<td>Router priority of neighbor and neighbor state.</td>
</tr>
<tr>
<td>State</td>
<td>OSPF state.</td>
</tr>
<tr>
<td>state changes</td>
<td>Number of state changes for this neighbor.</td>
</tr>
<tr>
<td>DR is</td>
<td>Neighbor ID of the designated router.</td>
</tr>
<tr>
<td>BDR is</td>
<td>Neighbor ID of the backup designated router.</td>
</tr>
<tr>
<td>Options</td>
<td>Hello packet options field contents. (E-bit only; possible values are 0 and 2; 2 indicates that the area is not a stub; 0 indicates that the area is a stub.).</td>
</tr>
<tr>
<td>LLS Options is 0x1 (LR)</td>
<td>Neighbor is NFS Cisco capable.</td>
</tr>
<tr>
<td>Dead timer</td>
<td>Time (in hh:mm:ss) to elapse before OSPF declares the neighbor dead.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Amount of time (in hh:mm:ss) that the OSPF neighbor has been up.</td>
</tr>
<tr>
<td>Number of DBD retrans</td>
<td>Number of re-sent database description packets.</td>
</tr>
</tbody>
</table>
Index and the remaining lines of this command give detailed information about flooding information received from the neighbor.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index</td>
<td>Index and the remaining lines of this command give detailed information about flooding information received from the neighbor.</td>
</tr>
</tbody>
</table>
**show ospf request-list**

To display the first ten link-state requests pending that the local router is making to the specified Open Shortest Path First (OSPF) neighbor and interface, use the `show ospf request-list` command in XR EXEC mode.

```
show ospf [process-name] [area-id] request-list [type interface-path-id] [neighbor-id]
```

**Syntax Description**

- `process-name` (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only information for the specified routing process is displayed.

- `area-id` (Optional) Area ID. If you do not specify an area, all areas are displayed.

- `type` Interface type. For more information, use the question mark (?) online help function.

- `interface-path-id` Physical interface or virtual interface.
  
  Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

  For more information about the syntax for the router, use the question mark (?) online help function.

- `neighbor-id` (Optional) IP address of the OSPF neighbor.

**Command Default**

All neighbors

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You might use this command when the databases of two neighboring routers are out of synchronization or if the adjacency does not form between them. Adjacency means that the routers synchronize their databases when they discover each other.

You can look at the list to determine if one router is trying to request a particular database update. Entries that are suspended in the list usually indicate that updates are not being delivered. One possible reason for this behavior is a maximum transmission unit (MTU) mismatch between the routers.

You might also look at this list to make sure it is not corrupted. The list should refer to database entries that actually exist.

Request list information is transient and normally the lists are empty.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>
The following is sample output from the `show ospf request-list` command:

```
RP/O/RP0/CPU0:router# show ospf request-list 10.0.124.4 tenGigE 3/0/0/0

Request Lists for OSPF pagent

Neighbor 10.0.124.4, interface tenGigE 3/0/0/0 address 10.3.1.2

<table>
<thead>
<tr>
<th>Type</th>
<th>LS ID</th>
<th>ADV RTR</th>
<th>Seq NO</th>
<th>Age</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>192.168.58.17</td>
<td>192.168.58.17</td>
<td>0x80000012</td>
<td>12</td>
<td>0x0036f3</td>
</tr>
<tr>
<td>2</td>
<td>192.168.58.68</td>
<td>192.168.58.17</td>
<td>0x80000012</td>
<td>12</td>
<td>0x0083f</td>
</tr>
</tbody>
</table>
```

This table describes the significant fields shown in the display.

**Table 50: show ospf request-list 10.0.124.4 tenGigE 3/0/0/0 Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Specific neighbor receiving the request list from the local router.</td>
</tr>
<tr>
<td>Interface</td>
<td>Specific interface over which the request list is being sent.</td>
</tr>
<tr>
<td>Address</td>
<td>Address of the interface over which the request list is being sent.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of link-state advertisement (LSA).</td>
</tr>
<tr>
<td>LS ID</td>
<td>Link-state ID of the LSA.</td>
</tr>
<tr>
<td>ADV RTR</td>
<td>IP address of the advertising router.</td>
</tr>
<tr>
<td>Seq NO</td>
<td>Sequence number of the LSA.</td>
</tr>
<tr>
<td>Age</td>
<td>Age of the LSA (in seconds).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Checksum of the LSA.</td>
</tr>
</tbody>
</table>
show ospf retransmission-list

To display the first ten link-state entries in the Open Shortest Path First (OSPF) retransmission list that the local router sends to the specified neighbor over the specified interface, use the show ospf retransmission-list command in XR EXEC mode.

```
show ospf [process-name] [area-id] retransmission-list [type interface-path-id] [neighbor-id]
```

Syntax Description

- **process-name** (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the router ospf command. If this argument is included, only information for the specified routing process is displayed.

- **interface-path-id** Physical interface or virtual interface.

  **Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.

- **neighbor-id** (Optional) IP address of the OSPF neighbor.

Command Default

All neighbors

Command Modes

XR EXEC mode

Command History

- **Release** 6.0 This command was introduced.

Usage Guidelines

You might use this command when the databases of two neighboring routers are out of synchronization or if the adjacency is not forming between them. Adjacency means that the routers synchronize their databases when they discover each other.

You can look at the list to determine if one router is trying to request a particular database update. Entries that appear to be suspended in the list usually indicate that updates are not being delivered. One possible reason for this behavior is a maximum transmission unit (MTU) mismatch between the routers.

You might also look at this list to make sure it is not corrupted. The list should refer to database entries that actually exist.

Retransmission list information is transient, and normally the lists are empty.

Task ID

- **Task ID** Operations

  - ospf read

Examples

The following is sample output from the show ospf retransmission-list command:
RP/0/RP0/CPU0:router#  show ospf retransmission-list 10.0.124.4  tenGigE 3/0/0/0
Neighbor 10.0.124.4, interface tenGigE 3/0/0/0 address 10.3.1.2

This table describes the significant fields shown in the display.

Table 51: show ospf retransmission-list 10.0.124.4 GigabitEthernet3/0/0/0 Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Specified neighbor receiving the retransmission list from the local router.</td>
</tr>
<tr>
<td>Interface</td>
<td>Specified interface over which the retransmission list is being sent.</td>
</tr>
<tr>
<td>Address</td>
<td>Address of the interface.</td>
</tr>
</tbody>
</table>
show ospf routes

To display the Open Shortest Path First (OSPF) topology table, use the **show ospf routes** command in XR EXEC mode.

```plaintext
show ospf [process-name] routes [{connected|external|local}] [prefix mask] [prefix /length] [multicast-intact] [backup-path]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>process-name</td>
<td>(Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the <code>router ospf</code> command. If this argument is included, only information for the specified routing process is displayed.</td>
</tr>
<tr>
<td>connected</td>
<td>(Optional) Displays connected routes.</td>
</tr>
<tr>
<td>external</td>
<td>(Optional) Displays routes redistributed from other protocols.</td>
</tr>
<tr>
<td>local</td>
<td>(Optional) Displays the local routes redistributed from the Routing Information Base (RIB).</td>
</tr>
<tr>
<td>prefix</td>
<td>(Optional) IP prefix, which limits output to a specific route. If the <code>prefix</code> argument is specified, either the <code>length</code> or <code>mask</code> argument is required.</td>
</tr>
<tr>
<td>mask</td>
<td>(Optional) IP address mask.</td>
</tr>
<tr>
<td>/ length</td>
<td>(Optional) Prefix length, which can be indicated as a slash (/) and number. For example, /8 indicates that the first eight bits in the IP prefix are network bits. If <code>length</code> is used, the slash is required.</td>
</tr>
</tbody>
</table>

**Command Default**

All route types

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **show ospf routes** command to display the OSPF private routing table (which contains only routes calculated by OSPF). If there is something wrong with a route in the RIB, then it is useful to check the OSPF copy of the route to determine if it matches the RIB contents. If it does not match, there is a synchronization problem between OSPF and the RIB. If the routes match and the route is incorrect, OSPF has made an error in its routing calculation.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>
show ospf routes command output with TI-LFA information

This is sample output from the `show ospf routes` command with the `backup-path` keyword that displays backup-path information, including TI-LFA:

```
RP/0/RP0/CPU0:router# show ospf routes 2.2.2.2/32 backup-path
Fri Apr 4 02:08:04.210 PDT

Topology Table for ospf 1 with ID 1.1.1.1

Codes: O - Intra area, O IA - Inter area
      O E1 - External type 1, O E2 - External type 2
      O N1 - NSSA external type 1, O N2 - NSSA external type 2

O  2.2.2.2/32, metric 3
   10.1.0.2, from 2.2.2.2, via tenGigE 0/0/0/7, path-id 1
      Backup path: TI-LFA, P node: 4.4.4.4, Labels: 16004, 123
      10.0.3.2, from 2.2.2.2, via tenGigE 0/0/0/3, protected bitmap 0x1
      Attributes: Metric: 104, SRLG Disjoint
```

This table describes the significant fields shown in the display.

### Table 52: show ospf route Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>OSPF route.</td>
</tr>
<tr>
<td>E</td>
<td>External Type 1 or 2 route.</td>
</tr>
<tr>
<td>N</td>
<td>NSSA Type 1 or 2</td>
</tr>
<tr>
<td>2.2.2.2/32</td>
<td>Network and subnet mask to which the local router has a route.</td>
</tr>
<tr>
<td>metric</td>
<td>Cost to reach network 10.3.1.0.</td>
</tr>
<tr>
<td>10.1.0.2</td>
<td>Next-hop router on the path to network 10.3.1.0.</td>
</tr>
<tr>
<td>from 2.2.2.2</td>
<td>Router ID 172.16.10.1 is the router that advertised this route.</td>
</tr>
<tr>
<td>via GigabitEthernet0/0/0/7</td>
<td>Packets destined for the given prefix (10.3.1.0/24) are sent over GigabitEthernet interface 0/0/0/7.</td>
</tr>
<tr>
<td>Backup path</td>
<td>Indicates the topology independent loop-free alternate backup path. Here, the backup path uses the P node 4.4.4.4.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospf routes` command:

```
RP/0/RP0/CPU0:router# show ospf routes

Topology Table for ospf 1 with ID 10.3.4.2

Codes:O - Intra area, O IA - Inter area
       O E1 - External type 1, O E2 - External type 2
       O N1 - NSSA external type 1, O N2 - NSSA external type 2
```
This table describes the significant fields shown in the display.

**Table 53: show ospf route Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>OSPF route.</td>
</tr>
<tr>
<td>E</td>
<td>External Type 1 or 2 route.</td>
</tr>
<tr>
<td>N</td>
<td>NSSA Type 1 or 2</td>
</tr>
<tr>
<td>10.3.1.0/24</td>
<td>Network and subnet mask to which the local router has a route.</td>
</tr>
<tr>
<td>metric</td>
<td>Cost to reach network 10.3.1.0.</td>
</tr>
<tr>
<td>10.3.4.1</td>
<td>Next-hop router on the path to network 10.3.1.0.</td>
</tr>
<tr>
<td>from 172.16.10.1</td>
<td>Router ID 172.16.10.1 is the router that advertised this route.</td>
</tr>
<tr>
<td>via GigabitEthernet 0/1/0/1</td>
<td>Packets destined for the given prefix (10.3.1.0/24) are sent over GigabitEthernet interface 0/1/0/1.</td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

**Table 54: show ospf route Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>OSPF route.</td>
</tr>
<tr>
<td>E2</td>
<td>External Type 2 route.</td>
</tr>
<tr>
<td>10.3.1.0/24</td>
<td>Network and subnet mask to which the local router has a route.</td>
</tr>
<tr>
<td>metric 1</td>
<td>Cost to reach network 10.3.1.0.</td>
</tr>
<tr>
<td>10.3.4.1</td>
<td>Next-hop router on the path to network 10.3.1.0.</td>
</tr>
<tr>
<td>from 172.16.10.1</td>
<td>Router ID 172.16.10.1 is the router that advertised this route.</td>
</tr>
<tr>
<td>via POS 0/1/0/1</td>
<td>Packets destined for the given prefix (10.3.1.0/24) are sent over POS interface 0/1/0/1.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospf routes` command with a process name of 100:
show ospf routes

RP/0/RP0/CPU0:router# show ospf 100 routes

Topology Table for ospf 100 with ID 172.23.54.14

Codes: O - Intra area, O IA - Inter area
  O E1 - External type 1, O E2 - External type 2
  O N1 - NSSA external type 1, O N2 - NSSA external type 2

  O  10.1.5.0/24, metric 1562
      10.1.5.14, directly connected, via tenGigE 0/3/0/3

  O IA 21.0.0.0/24, metric 1572
      10.1.5.12, from 172.23.54.12, via tenGigE 0/3/0/3

  O  10.0.0.0/24, metric 10
      10.0.0.12, directly connected, via tenGigE 0/2/0/3

This table describes the significant fields shown in the display.

Table 55: show ospf 100 route Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>OSPF route.</td>
</tr>
<tr>
<td>IA</td>
<td>Interarea route.</td>
</tr>
<tr>
<td>10.1.5.0/24</td>
<td>Network and subnet mask to which the local router has a route.</td>
</tr>
<tr>
<td>metric 1562</td>
<td>Cost to reach network 10.1.5.0.</td>
</tr>
<tr>
<td>10.1.5.14</td>
<td>Next-hop router on the path to network 10.1.5.0.</td>
</tr>
<tr>
<td>from 172.23.54.12</td>
<td>Router ID 172.23.54.12 is the router that advertised this route.</td>
</tr>
<tr>
<td>via GigabitEthernet 0/3/0/3</td>
<td>Packets destined for the given prefix (10.3.1.0/24) are sent over GigabitEthernet interface 0/3/0/3.</td>
</tr>
</tbody>
</table>

The following is sample output from the show ospf routes command with a prefix of 10.0.0.0 and a length of 24:

RP/0/RP0/CPU0:router# show ospf routes 10.0.0.0/24

Topology Table for ospf 100 with ID 172.23.54.14

Codes: O - Intra area, O IA - Inter area
  O E1 - External type 1, O E2 - External type 2
  O N1 - NSSA external type 1, O N2 - NSSA external type 2

  O IA 10.0.0.0/24, metric 1572
      10.1.5.12, from 172.23.54.12, via GigabitEthernet 0/3/0/3
This table describes the significant fields shown in the display.

**Table 56: show ospf route 10.0.0.0/24 Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>O</td>
<td>Route is an OSPF route.</td>
</tr>
<tr>
<td>IA</td>
<td>Route to network 10.0.0.0 is an interarea route.</td>
</tr>
<tr>
<td>10.0.0.0/24</td>
<td>Network and subnet mask to which the local router has a route.</td>
</tr>
<tr>
<td>metric 1572</td>
<td>Cost to reach network 10.0.0.0.</td>
</tr>
<tr>
<td>10.1.5.12</td>
<td>IP address of next-hop router on the path to network 10.0.0.0.</td>
</tr>
<tr>
<td>from 172.23.54.12</td>
<td>Router ID 172.23.54.12 is the router that advertised this route.</td>
</tr>
<tr>
<td>via GigabitEthernet 0/3/0/3</td>
<td>Packets destined for the given prefix (10.0.0.0/24) are sent over GigabitEthernet interface 0/3/0/3.</td>
</tr>
</tbody>
</table>
# show ospf sham-links

To display Open Shortest Path First (OSPF) sham-link information, use the `show ospf sham-links` command in XR EXEC mode.

```
show ospf [process-name] sham-links
```

**Syntax Description**

- `process-name` (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only information for the specified routing process is displayed.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show ospf sham-links` command to display OSPF sham-link information.

**Task ID**

```
Task | Operations ID
----|---------------
ospf | read
```

**Examples**

The following is sample output from the `show ospf sham-links` command:

```
RP/0/RP0/CPU0:router# show ospf 1 vrf vrf_1 sham-links

Sham Links for OSPF 1, VRF vrf_1
Sham Link OSPF_SL0 to address 10.0.0.3 is up
Area 0, source address 10.0.0.1
IfIndex = 185
Run as demand circuit
DoNotAge LSA allowed, Cost of using 1
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 00:00:04
Adjacency State FULL (Hello suppressed)
Number of DBD retrans during last exchange 0
Index 2/2, retransmission queue length 0, number of retransmission 0
First 0(0)/0(0) Next 0(0)/0(0)
Last retransmission scan length is 0, maximum is 0
Last retransmission scan time is 0 msec, maximum is 0 msec
Keychain-based authentication enabled
Key id used is 2
```
This table describes the significant fields shown in the display.

**Table 57: show ospf sham-links Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sham Link OSPF_SL0 to address</td>
<td>Address of the destination endpoint of the sham link.</td>
</tr>
<tr>
<td>IfIndex</td>
<td>ifindex associated with the sham link.</td>
</tr>
<tr>
<td>Run as demand circuit</td>
<td>Sham link is treated as a demand circuit.</td>
</tr>
<tr>
<td>DoNotAge LSA allowed</td>
<td>DoNotAge LSAs are allowed to be flooded over the sham link.</td>
</tr>
<tr>
<td>Cost of using</td>
<td>Sham-link cost.</td>
</tr>
<tr>
<td>Transmit Delay</td>
<td>Sham-link transmit delay.</td>
</tr>
<tr>
<td>State</td>
<td>Sham-link interface state.</td>
</tr>
<tr>
<td>Timer intervals configured</td>
<td>Various sham-link interface-related timers.</td>
</tr>
<tr>
<td>Hello due in</td>
<td>Time before the next Hello is sent over the sham link.</td>
</tr>
<tr>
<td>Adjacency State</td>
<td>State of the adjacency with the neighbor over the sham link.</td>
</tr>
<tr>
<td>Number of DBD retrans during last exchange</td>
<td>Number of DBD retransmissions during the last exchange over the sham link.</td>
</tr>
<tr>
<td>Index</td>
<td>Area flood index.</td>
</tr>
<tr>
<td>retransmission queue length</td>
<td>Retransmission queue length on the sham link.</td>
</tr>
<tr>
<td>number of retransmission</td>
<td>Number of retransmissions over the sham-link interface.</td>
</tr>
<tr>
<td>First</td>
<td>First flood information.</td>
</tr>
<tr>
<td>Next</td>
<td>Next flood information.</td>
</tr>
<tr>
<td>Last retransmission scan length is</td>
<td>Last retransmission scan length on the sham-link interface.</td>
</tr>
<tr>
<td>maximum is</td>
<td>Maximum retransmission scan length on the sham-link interface.</td>
</tr>
<tr>
<td>Last retransmission scan time is</td>
<td>Last retransmission scan time on the sham-link interface.</td>
</tr>
<tr>
<td>maximum is 0 msec</td>
<td>Maximum retransmission scan time on the sham-link interface.</td>
</tr>
<tr>
<td>Keychain-based authentication enabled</td>
<td>Keychain-based authentication is enabled.</td>
</tr>
<tr>
<td>Key id used is</td>
<td>Key ID used.</td>
</tr>
</tbody>
</table>
show ospf summary-prefix

To display Open Shortest Path First (OSPF) aggregated summary address information, use the `show ospf summary-prefix` command in XR EXEC mode.

```
show ospf [process-name] summary-prefix
```

**Syntax Description**

- `process-name` (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only information for the specified routing process is displayed.

**Command Default**

All summary prefixes

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show ospf summary-prefix` command if you configured summarization of external routes with the `summary-prefix` command and you want to display configured summary addresses.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show ospf summary-prefix` command:

```
RP/0/RP0/CPU0:router# show ospf summary-prefix
OSPF Process 1, summary-prefix
10.1.0.0/255.255.0.0 Metric 20, Type 2, Tag 0
```

This table describes the significant fields shown in the display.

**Table 58: show ospf summary-prefix Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.1.0.0/255.255.0.0</td>
<td>Summary address designated for a range of addresses. The IP subnet mask used for the summary route.</td>
</tr>
<tr>
<td>Metric</td>
<td>Metric used to advertise the summary routes.</td>
</tr>
<tr>
<td>Type</td>
<td>External link-state advertisements (LSA) metric type.</td>
</tr>
</tbody>
</table>
### Tag

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag</td>
<td>Tag value that can be used as a “match” value for controlling redistribution through route maps.</td>
</tr>
</tbody>
</table>
show ospf virtual-links

To display parameters and the current state of Open Shortest Path First (OSPF) virtual links, use the `show ospf virtual-links` command in XR EXEC mode.

```
show ospf [process-name] virtual-links
```

**Syntax Description**

process-name  (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospf` command. If this argument is included, only information for the specified routing process is displayed.

**Command Default**

All virtual links

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**

Use the `show ospf virtual-links` command to display useful information for debugging OSPF routing operations.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show ospf virtual-links` command:

```
RP/0/RP0/CPU0:router# show ospf virtual-links

Virtual Link to router 172.31.101.2 is up
Transit area 0.0.0.1, via interface GigabitEthernet 0/3/0/0, Cost of using 10
Transmit Delay is 1 sec, State POINT_TO_POINT
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 0:00:08
Adjacency State FULL
```

This table describes the significant fields shown in the display.

**Table 59: show ospf virtual-links Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Link to router 172.31.101.2 is up</td>
<td>OSPF neighbor and whether the link to that neighbor is up or down.</td>
</tr>
<tr>
<td>Transit area 0.0.0.1</td>
<td>Transit area through which the virtual link is formed.</td>
</tr>
<tr>
<td><strong>Field</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------------------------------------------------------</td>
</tr>
<tr>
<td>via interface GigabitEthernet 0/3/0/0</td>
<td>Interface through which the virtual link is formed.</td>
</tr>
<tr>
<td>Cost of using us using 10</td>
<td>Cost of reaching the OSPF neighbor through the virtual link.</td>
</tr>
<tr>
<td>Transmit Delay is 1 sec</td>
<td>Transmit delay (in seconds) on the virtual link.</td>
</tr>
<tr>
<td>State POINT_TO_POINT</td>
<td>State of the OSPF neighbor.</td>
</tr>
<tr>
<td>Timer intervals</td>
<td>Various timer intervals (in seconds) configured for the link.</td>
</tr>
<tr>
<td>Hello due in 0:00:08</td>
<td>When the next hello message is expected from the neighbor (in hh:mm:ss).</td>
</tr>
<tr>
<td>Adjacency State FULL</td>
<td>Adjacency state between the neighbors.</td>
</tr>
</tbody>
</table>
show protocols (OSPF)

To display information about the OSPFv2 processes running on the router, use the `show protocols` command in XR EXEC mode.

```
show protocols [{afi-all|ipv4|ipv6}] [{all|protocol}]
```

### Syntax Description

<table>
<thead>
<tr>
<th>afi-all</th>
<th>(Optional) Specifies all address families.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies an IPv4 address family.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies an IPv6 address family.</td>
</tr>
<tr>
<td>all</td>
<td>(Optional) Specifies all protocols for a given address family.</td>
</tr>
<tr>
<td>protocol</td>
<td>(Optional) Specifies a routing protocol. For the IPv4 address family, the options are:</td>
</tr>
<tr>
<td></td>
<td>• bgp</td>
</tr>
<tr>
<td></td>
<td>• isis</td>
</tr>
<tr>
<td></td>
<td>• ospf</td>
</tr>
<tr>
<td></td>
<td>• rip</td>
</tr>
<tr>
<td></td>
<td>For the IPv6 address family, the options are:</td>
</tr>
<tr>
<td></td>
<td>• bgp</td>
</tr>
<tr>
<td></td>
<td>• isis</td>
</tr>
<tr>
<td></td>
<td>• ospfv3</td>
</tr>
</tbody>
</table>

### Command Default

No default behavior or value

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

No specific guidelines impact the use of this command.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

### Examples

The following is an OSPF configuration and the resulting `show protocols ospf` display:

```
RP/0/RP0/CPU0:router#show running router ospf 1
```
router ospf 1
  router-id Loopback0
  nsf
  redistribute connected
  redistribute isis 3
  area 0
  mpls traffic-eng
  interface Loopback0
  !
  interface Loopback1
  !
  interface Loopback2
  !
  interface tenGigE 0/3/0/0
  !
  interface tenGigE 0/3/0/1
  !
  interface tenGigE 0/3/0/2
  !
  interface tenGigE 0/3/0/3
  !
  mpls traffic-eng router-id Loopback0

RP/0/RP0/CPU0:router# show protocols ospf
Routing Protocol OSPF 1
  Router Id: 55.55.55.55
  Distance: 110
  Non-Stop Forwarding: Enabled
  Redistribution:
    connected
    isis 3
  Area 0
    MPLS/TE enabled
    tenGigE 0/3/0/3
    tenGigE 0/3/0/2
    tenGigE 0/3/0/1
    tenGigE 0/3/0/0
    Loopback2
    Loopback0

This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router Id</td>
<td>ID of the router for this configuration.</td>
</tr>
<tr>
<td>Distance</td>
<td>Administrative distance of OSPF routes relative to routes from other protocols.</td>
</tr>
<tr>
<td>Non-Stop Forwarding</td>
<td>Status of nonstop forwarding.</td>
</tr>
<tr>
<td>Redistribution</td>
<td>Lists the protocols that are being redistributed.</td>
</tr>
<tr>
<td>Area</td>
<td>Information about the current area including list of interfaces and the status of Multiprotocol Label Switching traffic engineering (MPLS TE).</td>
</tr>
</tbody>
</table>
**snmp context (OSPF)**

To specify an SNMP context for an OSPF instance, use the `snmp context` command in XR Config mode or in VRF configuration mode. To remove the SNMP context, use the `no` form of this command.

```
  snmp context  context_name
  no snmp context  context_name
```

**Syntax Description**
- `context_name` Specifies name of the SNMP context for OSPF instance.

**Command Default**
SNMP context is not specified.

**Command Modes**
- XR Config mode
- VRF configuration

**Command History**
- **Release** 6.0  This command was introduced.

**Usage Guidelines**
The `snmp-server` commands need to be configured to perform SNMP request for the OSPF instance. Refer `SNMP Server Commands` module in `System Management Command Reference` for information on using the `snmp-server` commands.

**Note**
To map an SNMP context with a protocol instance, topology or VRF entity, use the `snmp-server context mapping` command. However, the `feature` option of this command does not work with OSPF protocol.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to configure an SNMP context `foo` for OSPF instance `100`:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospf 100
RP/0/RP0/CPU0:router(config-ospf)#snmp context foo
```

This example shows how to configure `snmp-server` commands to be used with the `snmp context` command:

```
RP/0/RP0/CPU0:router(config)#snmp-server host 10.0.0.2 traps version 2c public udp-port 1620
RP/0/RP0/CPU0:router(config)#snmp-server community public RW
RP/0/RP0/CPU0:router(config)#snmp-server contact foo
```
This is a sample SNMP context configuration for OSPF instance 100:

```
RP/0/RP0/CPU0:router(config)#snmp-server community-map public context foo

snmp-server host 10.0.0.2 traps version 2c public udp-port 1620
snmp-server community public RW
snmp-server contact foo

snmp-server community-map public context foo

router ospf 100
  router-id 2.2.2.2
  bfd fast-detect
  nsf cisco
  snmp context foo
  area 0
    interface Loopback1
    !
  !
  area 1
    interface tenGigE 0/2/0/1
    demand-circuit enable
    !
    interface tenGigE 0/3/0/0
    !
    interface tenGigE 0/3/0/1
    !
    !
```

Routing Command Reference for Cisco NCS 5000 Series Routers
The `snmp trap (OSPF)` command is used to configure SNMP traps for an OSPF instance. To enable SNMP traps for an OSPF instance, use the `snmp trap` command in VRF configuration mode. To disable SNMP traps for the OSPF instance, use the `no` form of this command.

**Syntax Description**

This command has no keywords or arguments.

**Command Default**

Disabled.

**Command Modes**

VRF configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to enable SNMP trap for OSPF instance 100 under VRF `vrf-1`:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospf 100
RP/0/RP0/CPU0:router(config-ospf)#vrf vrf-1
RP/0/RP0/CPU0:router(config-ospf-vrf)#snmp trap
```
To control the number of traps that OSPF sends by configuring window size and the maximum number of traps during that window, use the `snmp trap rate-limit` command in XR Config mode. To disable configuring the window size and maximum number of traps during the window, use the `no` form of this command.

```
snmp trap rate-limit window-size max-num-traps
no snmp trap rate-limit window-size max-num-traps
```

**Syntax Description**

- `window-size` Specifies the trap rate limit sliding window size.
- `max-num-traps` Specifies the maximum number of traps sent in window time.

**Command Default**

None

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the trap rate limit sliding window size to 30 and the maximum number of traps sent to 100:

```
RP/0/RP0/CPU0:router(config)#router ospf 100
RP/0/RP0/CPU0:router(config-ospf)#snmp trap rate-limit 30 100
```
**spf prefix-priority (OSPFv2)**

To prioritize OSPFv2 prefix installation into the global Routing Information Base (RIB) during Shortest Path First (SPF) run, use the `spf prefix-priority` command in router configuration mode. To return to the system default value, use the `no` form of this command.

```
spf prefix-priority route-policy policy-name
no spf prefix-priority route-policy policy-name
```

**Syntax Description**

- `route-policy policy-name` Specifies the route policy to apply to OSPFv2 prefix prioritization.

**Note**

If SPF prefix prioritization is configured, /32 prefixes are no longer preferred by default. To retain the /32 prefixes in higher-priority queues, define the route-policy accordingly.

---

**Command Default**

SPF prefix prioritization is disabled.

**Command Modes**

OSPF router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

SPF prefix prioritization is disabled, by default. In disabled mode, the /32 prefixes are installed into the global RIB before other prefixes.

If SPF prefix prioritization is enabled, routes are matched against the route-policy criteria and are assigned to the appropriate priority queue based on the spf-priority set. Unmatched prefixes, including the /32 prefixes, are placed in the low-priority queue.

If all /32 prefixes are desired in the high-priority queue or medium-priority queue, configure the following single route map:

```
prefix-set ospf-medium-prefixes
  0.0.0.0/0 ge 32
end-set
```

**Examples**

The following example shows how to configure OSPFv2 SPF prefix prioritization:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# prefix-set ospf-critical-prefixes
RP/0/RP0/CPU0:router(config-pfx)# 66.0.0.0/16
```
RP/0/RP0/CPU0:router(config-pfx)# end-set
RP/0/RP0/CPU0:router(config)# route-policy ospf-spf-priority
RP/0/RP0/CPU0:router(config-rpl)# if destination in ospf-critical-prefixes then set spf-priority critical
endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# router-id 66.0.0.1
RP/0/RP0/CPU0:router(config-ospf)# spf prefix-priority route-policy ospf-spf-priority
stub (OSPF)

To define an area as a stub area, use the stub command in area configuration mode. To disable this function, use the no form of this command.

```
stub [no-summary]
no stub
```

**Syntax Description**

- **no-summary** (Optional) Prevents an Area Border Router (ABR) from sending summary link advertisements into the stub area.

**Command Default**

No stub area is defined.

**Command Modes**

Area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You must configure the stub command on all routers in the stub area.

Use the default-cost command on the ABR of a stub area to specify the cost of the default route advertised into the stub area by the ABR.

To further reduce the number of link-state advertisements (LSAs) sent into a stub area, you can configure the no-summary keyword on the ABR to prevent it from sending summary LSAs (LSA Type 3) into the stub area.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to assign a default cost of 20 to stub network 10.0.0.0:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
RP/0/RP0/CPU0:router(config-ospf)# area 10.0.0.0
RP/0/RP0/CPU0:router(config-ospf-ar)# stub
RP/0/RP0/CPU0:router(config-ospf-ar)# default-cost 20
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/3/0/3
```
**summary-prefix (OSPF)**

To create aggregate addresses for routes being redistributed from another routing protocol into the Open Shortest Path First (OSPF) protocol, use the `summary-prefix` command in the appropriate mode. To stop summarizing redistributed routes, use the `no` form of the command.

```
summary-prefix address mask [not-advertise|tag tag]
no summary-prefix address mask
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>address</td>
<td>Summary address designated for a range of addresses.</td>
</tr>
<tr>
<td>mask</td>
<td>IP subnet mask used for the summary route.</td>
</tr>
<tr>
<td>not-advertise</td>
<td>(Optional) Suppresses summary routes that match the address and mask pair from being advertised.</td>
</tr>
<tr>
<td>tag tag</td>
<td>(Optional) Tag value that can be used as a “match” value for controlling redistribution through route policies.</td>
</tr>
</tbody>
</table>

**Command Default**

When this command is not used, specific addresses are created for each route from another route source being distributed into the OSPF protocol.

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `summary-prefix` command to cause an OSPF Autonomous System Boundary Router (ASBR) to advertise one external route as an aggregate for all redistributed routes that are covered by the address. This command summarizes only routes from other routing protocols that are being redistributed into OSPF.

You can use this command multiple times to summarize multiple groups of addresses. The metric used to advertise the summary is the lowest metric of all the more specific routes. This command helps reduce the size of the routing table.

If you want to summarize routes between OSPF areas, use the `range` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, summary address 10.1.0.0 includes address 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. Only the address 10.1.0.0 is advertised in an external link-state advertisement.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
```
RP/0/RP0/CPU0:router(config-ospf) # summary-prefix 10.1.0.0 255.255.0.0
timers lsa group-pacing

To change the interval at which Open Shortest Path First (OSPF) link-state advertisements (LSAs) are collected into a group and refreshed, checksummed, or aged, use the `timers lsa group-pacing` command in the appropriate mode. To restore the default value, use the `no` form of this command.

```
    timers lsa group-pacing seconds
    no timers lsa group-pacing
```

**Syntax Description**

- `seconds` Interval (in seconds) at which LSAs are grouped and refreshed, checksummed, or aged. Range is 10 seconds to 1800 seconds.

**Command Default**

- `seconds`: 240 seconds

**Command Modes**

- Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**

OSPF LSA group pacing is enabled by default. For typical customers, the default group pacing interval for refreshing, checksumming, and aging is appropriate and you need not configure this feature.

The duration of the LSA group pacing is inversely proportional to the number of LSAs the router is handling. For example, if you have approximately 10,000 LSAs, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

**Task ID**

- `ospf` read, write

**Examples**

The following example shows how to change the OSPF pacing between LSA groups to 60 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# timers lsa group-pacing 60
```
timers lsa min-arrival

To limit the frequency that new instances of any particular Open Shortest Path First (OSPF) link-state advertisements (LSAs) can be accepted during flooding, use the `timers lsa min-arrival` command in the appropriate mode. To restore the default value, use the `no` form of this command.

`timers lsa min-arrival milliseconds`

`no timers lsa min-arrival`

**Syntax Description**

<table>
<thead>
<tr>
<th><code>milliseconds</code></th>
<th>Minimum interval (in milliseconds) between accepting same LSA.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Range is 0 to 600000 milliseconds.</td>
</tr>
</tbody>
</table>

**Command Default**

`milliseconds`: 100 milliseconds

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to change the minimum interval between accepting the same LSA to 2 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# timers lsa min-arrival 2
```
timers throttle lsa all (OSPF)

To modify the Open Shortest Path First (OSPF) link-state advertisement (LSA) throttling, use the **timers throttle lsa all** command in the appropriate mode. To revert LSA throttling to default settings, use the **no** form of this command.

```
timers throttle lsa all start-interval hold-interval max-interval
no timers throttle lsa all
```

**Syntax Description**

- **start-interval**: Delay to generate first occurrence of LSA in milliseconds. Range is 0 to 600000 milliseconds.
- **hold-interval**: Minimum delay between originating the same LSA in milliseconds. Range is 1 to 600000 milliseconds.
- **max-interval**: Maximum delay between originating the same LSA in milliseconds. Range is 1 to 600000 milliseconds.

**Command Default**

- **start-interval**: 50 milliseconds
- **hold-interval**: 200 milliseconds
- **max-interval**: 5000 milliseconds

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The **lsa-start** time is the delay before flooding the first instance of an LSA. The **lsa-hold** interval is the minimum time to elapse before flooding an updated instance of an LSA. The **lsa-max-wait** time is the maximum time that can elapse before flooding an updated instance of an LSA.

For quick convergence, use smaller times for the **lsa-start** time and **lsa-hold** interval. However, in relatively large networks, this may result in a large number of LSAs being flooded in a relatively short time. A balance with the **lsa-start** time and **lsa-hold** interval can be iteratively arrived at for the size of your network. The **lsa-max-wait** time can be used to ensure that OSPF reconverges within a reasonable amount of time.

**Note**

LSA throttling is always enabled. You can change the timer values with the **timers throttle lsa all** command or specify the **no** keyword to revert back to the default settings.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to change the start, hold, and maximum wait interval values to 500, 1000, and 90,000 milliseconds, respectively:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# timers throttle lsa all 500 1000 90000
```

The following example is output from the `show ospf` command that displays the modified LSA throttle settings:

```
RP/0/RP0/CPU0:router# show ospf
Routing Process "ospf 1" with ID 1.1.1.1
Supports only single TOS(TOS0) routes
Supports opaque LSA
It is an area border router
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPFs 10000 msecs
Maximum wait time between two consecutive SPFs 10000 msecs
Initial LSA throttle delay 500 msecs
Minimum hold time for LSA throttle 1000 msecs
Maximum wait time for LSA throttle 90000 msecs
Minimum LSA interval 1000 msecs. Minimum LSA arrival 1 secs
Maximum number of configured interfaces 255
Number of external LSA 0. Checksum Sum 00000000
Number of opaque AS LSA 0. Checksum Sum 00000000
Number of DCbitless external and opaque AS LSA 0
Number of DoNotAge external and opaque AS LSA 0
Number of areas in this router is 2. 2 normal 0 stub 0 nssa
External flood list length 0
Non-Stop Forwarding enabled
Area BACKBONE(0) (Inactive)
   Number of interfaces in this area is 2
   SPF algorithm executed 8 times
   Number of LSA 2. Checksum Sum 0x01ba83
   Number of opaque link LSA 0. Checksum Sum 00000000
   Number of DCbitless LSA 0
   Number of indication LSA 0
   Number of DoNotAge LSA 0
   Flood list length 0
Area 1
   Number of interfaces in this area is 1
   SPF algorithm executed 9 times
   Number of LSA 2. Checksum Sum 0x0153ea
   Number of opaque link LSA 0. Checksum Sum 00000000
   Number of DCbitless LSA 0
   Number of indication LSA 0
   Number of DoNotAge LSA 0
   Flood list length 0
```
timers throttle spf (OSPF)

To modify the Open Shortest Path First (OSPF) shortest path first (SPF) throttling, use the `timers throttle spf` command in the appropriate mode. To revert SPF throttling to default settings, use the `no` form of this command.

```
   timers throttle spf  spf-start  spf-hold  spf-max-wait
   no timers throttle spf
```

**Syntax Description**
- **spf-start**: Initial SPF schedule delay (in milliseconds). Range is 1 to 600000 milliseconds.
- **spf-hold**: Minimum hold time (in milliseconds) between two consecutive SPF calculations. Range is 1 to 600000 milliseconds.
- **spf-max-wait**: Maximum wait time (in milliseconds) between two consecutive SPF calculations. Range is 1 to 600000 milliseconds.

**Command Default**
- `spf-start`: 50 milliseconds
- `spf-hold`: 200 milliseconds
- `spf-max-wait`: 5000 milliseconds

**Command Modes**
- Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The `spf-start` time is the delay before running SPF for the first time. The `spf-hold` interval is the minimum time to elapse between subsequent SPF runs. The `spf-max-wait` time is the maximum time that can elapse before running SPF again.

**Tip**

Setting a low `spf-start` time and `spf-hold` time causes routing to switch to the alternate path more quickly if there is a failure; however, it consumes more CPU processing time.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to change the start, hold, and maximum wait interval values to 5, 1000, and 90000 milliseconds, respectively:

```
RP/0/RP0/CPU0:router# configure
```
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# timers throttle spf 5 1000 90000
transmit-delay (OSPF)

To set the estimated time required to send a link-state update packet on the interface, use the `transmit-delay` command in the appropriate mode. To return to the default value, use the `no` form of this command.

```
transmit-delay seconds
no transmit-delay seconds
```

**Syntax Description**
- `seconds`: Time (in seconds) required to send a link-state update. Range is 1 to 65535 seconds.

**Command Default**
- `seconds`: 1 second

**Command Modes**
- XR Config mode
- Area configuration
- Interface configuration
- Virtual-link configuration
- Multi-area configuration
- Sham-link configuration

**Command History**
- **Release 6.0**: This command was introduced.

**Usage Guidelines**
Link-state advertisements (LSAs) in the update packet must have their ages incremented by the amount specified in the `seconds` argument before transmission. The value assigned should take into account the transmission and propagation delays for the interface.

If the delay is not added before transmission over a link, the time in which the LSA propagates over the link is not considered. This setting has significance only on very low-speed networks not supported in Cisco IOS XR software or on networks such as satellite circuits that incur a very long (greater than one second) delay time.

**Examples**
The following example shows how to configure a transmit delay for interface tenGigE 0/3/0/0:

```
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# area 0
RP/0/RP0/CPU0:router(config-ospf-ar)# interface tenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-ospf-ar-if)# transmit-delay 3
```
transmit-delay (OSPF)
virtual-link (OSPF)

To define an Open Shortest Path First (OSPF) virtual link, use the `virtual-link` command in area configuration mode. To remove a virtual link, use the `no` form of this command.

```
virtual-link router-id
no virtual-link router-id
```

**Syntax Description**

- `router-id` Router ID associated with the virtual link neighbor. The router ID appears in the `show ospf` command display. The router ID can be any 32-bit router ID value specified in four-part, dotted-decimal notation.

**Command Default**

No virtual links are defined.

**Command Modes**

Area configuration

**Command History**

- **Release** 6.0  This command was introduced.

**Usage Guidelines**

All areas in an OSPF autonomous system must be physically connected to the backbone area (area 0). In some cases in which this physical connection is not possible, you can use a virtual link to connect to the backbone through a nonbackbone area. You can also use virtual links to connect two parts of a partitioned backbone through a nonbackbone area. The area through which you configure the virtual link, known as a transit area, must have full routing information. The transit area cannot be a stub or not-so-stubby area.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to establish a virtual link with default values for all optional parameters:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
RP/0/RP0/CPU0:router(config-ospf)# area 10.0.0.0
RP/0/RP0/CPU0:router(config-ospf-ar)# virtual-link 10.3.4.5
RP/0/RP0/CPU0:router(config-ospf-ar-vl)#
```

The following example shows how to establish a virtual link with clear text authentication called mykey:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 201
RP/0/RP0/CPU0:router(config-ospf)# area 10.0.0.0
```
virtual-link (OSPF)

```
RP/0/RP0/CPU0:router(config-ospf-ar)# virtual-link 10.3.4.5
RP/0/RP0/CPU0:router(config-ospf-ar-vl)# authentication-key 0 mykey
```
vrf (OSPF)

To configure an Open Shortest Path First (OSPF) VPN routing and forwarding (VRF) instance, use the vrf command in XR Config mode. To terminate an OSPF VRF, use the no form of this command.

Note

VRF is not supported.

```
vrf vrf-name
no vrf vrf-name
```

**Syntax Description**

`vrf-name` Identifier of an OSPF VRF. The `vrf-name` argument can be specified as an arbitrary string. The strings “default” and “all” are reserved VRF names.

**Command Default**

No OSPF VRF is defined.

**Command Modes**

Router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the vrf command to explicitly configure a VRF. Commands configured under the VRF configuration mode (such as the interface [OSPF] and authentication commands) are automatically bound to that VRF.

To modify or remove the VRF, the vrf-id argument format must be the same as the format used when creating the area.

Note

To remove the specified VRF from the router configuration, use the no vrf vrf-id command. The no vrf vrf-id command removes the VRF and all VRF options, such as authentication, default-cost, nssa, range, stub, virtual-link, and interface.

To avoid possibly having the router ID change under a VRF, explicitly configure the router ID using the router-id command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure VRF vrf1 and tenGigE interface 0/2/0/0. The tenGigE interface 0/2/0/0 is bound to VRF vrf1 automatically.
configure
router ospf 1
vrf vrf1
area area1
interface tenGigE 0/2/0/0
OSPFv3 Commands

This module describes the commands used to configure and monitor the IP Version 6 (IPv6) Open Shortest Path First Version 3 (OSPFv3) routing protocol.

For detailed information about OSPFv3 concepts, configuration tasks, and examples, see the Implementing OSPF on Cisco NCS 5000 Series Routers module in the Routing Configuration Guide for Cisco NCS 5000 Series Routers.

Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

- address-family (OSPFv3), on page 326
- area (OSPFv3), on page 327
- authentication (OSPFv3), on page 328
- auto-cost (OSPFv3), on page 330
- capability vrf-lite (OSPFv3), on page 331
- clear ospfv3 process, on page 332
- clear ospfv3 redistribution, on page 333
- clear ospfv3 routes, on page 334
- clear ospfv3 statistics, on page 335
- cost (OSPFv3), on page 336
- database-filter all out (OSPFv3), on page 337
- dead-interval (OSPFv3), on page 338
- default-cost (OSPFv3), on page 340
- default-information originate (OSPFv3), on page 341
- default-metric (OSPFv3), on page 343
- demand-circuit (OSPFv3), on page 344
- distance ospfv3, on page 346
- distribute-list prefix-list in, on page 348
- distribute-list prefix-list out, on page 350
- domain-id (OSPFv3), on page 352
- encryption, on page 353
- flood-reduction (OSPFv3), on page 355
- graceful-restart (OSPFv3), on page 356
• hello-interval (OSPFv3), on page 357
• instance (OSPFv3), on page 359
• interface (OSPFv3), on page 361
• log adjacency changes (OSPFv3), on page 363
• maximum interfaces (OSPFv3), on page 364
• maximum paths (OSPFv3), on page 365
• maximum redistributed-prefixes (OSPFv3), on page 366
• mtu-ignore (OSPFv3), on page 368
• neighbor (OSPFv3), on page 370
• network (OSPFv3), on page 372
• nssa (OSPFv3), on page 374
• ospfv3 name-lookup, on page 376
• packet-size (OSPFv3), on page 377
• passive (OSPFv3), on page 378
• priority (OSPFv3), on page 380
• protocol shutdown (OSPFv3), on page 381
• range (OSPFv3), on page 382
• redistribute (OSPFv3), on page 384
• retransmit-interval (OSPFv3), on page 388
• router-id (OSPFv3), on page 390
• router ospfv3, on page 392
• sham-link (OSPFv3), on page 393
• show ospfv3, on page 394
• show ospfv3 border-routers, on page 400
• show ospfv3 database, on page 402
• show ospfv3 flood-list, on page 414
• show ospfv3 interface, on page 416
• show ospfv3 message-queue, on page 419
• show ospfv3 neighbor, on page 421
• show ospfv3 request-list, on page 427
• show ospfv3 retransmission-list, on page 429
• show ospfv3 routes, on page 431
• show ospfv3 statistics rib-thread, on page 433
• show ospfv3 summary-prefix, on page 435
• show ospfv3 virtual-links, on page 437
• show protocols (OSPFv3), on page 439
• snmp context (OSPFv3), on page 441
• snmp trap (OSPFv3), on page 443
• snmp trap rate-limit (OSPFv3), on page 444
• spf prefix-priority (OSPFv3), on page 445
• stub (OSPFv3), on page 446
• stub-router, on page 447
• summary-prefix (OSPFv3), on page 449
• timers lsa arrival, on page 451
• timers pacing flood, on page 452
• timers pacing lsa-group, on page 453
• timers pacing retransmission, on page 455
• timers throttle lsa all (OSPFv3), on page 456
• timers throttle spf (OSPFv3), on page 458
• trace (OSPFv3), on page 460
• transmit-delay (OSPFv3), on page 462
• virtual-link (OSPFv3), on page 463
address-family (OSPFv3)

To enter address family configuration mode for Open Shortest Path First Version 3 (OSPFv3), use the `address-family` command in the router ospfv3 configuration mode. To disable address family configuration mode, use the `no` form of this command.

```
address-family ipv6 [unicast]
no address-family ipv6 [unicast]
```

**Syntax Description**
- `ipv6` Specifies IP Version 6 (IPv6) address prefixes.
- `unicast` (Optional) Specifies unicast address prefixes.

**Command Default**
An address family is not specified.

**Command Modes**
Router ospfv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to configure the OSPFv3 router process with IPv6 unicast address prefixes:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# address-family ipv6 unicast
```
area (OSPFv3)

To configure an Open Shortest Path First Version 3 (OSPFv3) area, use the area command in an appropriate configuration mode. To remove an OSPFv3 area, use the no form of this command.

```
area area-id
no area area-id
```

**Syntax Description**

| Identifier of an OSPFv3 area. The area-id argument can be specified as either a decimal value or as an IPv4 address. |

**Command Default**

No OSPFv3 areas are defined.

**Command Modes**

Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

An area must be explicitly configured with the area command.

Use the area command to place the router in area configuration mode (prompt: config-router-ar), from which you can configure area-specific settings. Commands configured under this mode (such as the interface command) are automatically bound to that area.

```
<Note>
To remove the specified OSPFv3 area from the router ospfv3 configuration, use the no area area-id command. The no area area-id command removes the OSPFv3 area including all OSPFv3 area options, and all the OSPFv3 interfaces and interface options that are configured under the area.
</Note>
```

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure area 0 for OSPFv3 process 1. The tenGigE 0/1/0/1 interface also is configured:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/1/0/1
```
authentication (OSPFv3)

To enable plain text, Message Digest 5 (MD5) authentication, or null authentication for an Open Shortest Path First Version 3 (OSPFv3) interface, use the `authentication` command in an appropriate configuration mode. To remove such authentication, use the `no` form of this command.

```
authentication {ipsec spi spi-value | md5|sha1} [[clear|password] password|disable}
no authentication
```

**Syntax Description**

- `ipsec` Specifies IP Security (IPSec).
  
  IPSec supported only for OSPFv3.

- `spi spi-value` Specifies a security policy index (SPI) value. Range is 256 to 4294967295.

- `md5` Enables Message Digest 5 (MD5) authentication.

- `sha1` Enables SHA1 authentication.

- `clear` (Optional) Specifies that the key be unencrypted.

- `password` (Optional) Specifies that the key be encrypted using a two-way algorithm.

- `password` Any contiguous string that can be entered from the keyboard.

- `disable` Disables authentication for OSPFv3 packets.

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the authentication parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the authentication parameter specified for the process.

If this command is not specified at any level, then the interface does not use authentication.

**Command Modes**

- Interface configuration
  
  Area configuration
  
  Router configuration
  
  Virtual-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `authentication` command to specify an authentication type for the interface, which overrides the authentication specified for the area to which this interface belongs. If this command is not included in the configuration file, the authentication configured in the area to which the interface belongs is assumed (as specified by the area `authentication` command).
The authentication type and password must be the same for all OSPFv3 interfaces that are to communicate with each other through OSPFv3.

**Note**
- If OSPFv3 is configured along with IPsec authentication, then it is likely that adjacencies may flap on a Route Processor Fail Over (RPFO) even when NSR and/or Graceful Restart is enabled.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to enable MD5 authentication:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospfv3 201
RP/0/RP0/CPU0:router(config-ospfv3)# router-id 10.1.1.1
RP/0/RP0/CPU0:router(config-ospfv3)# authentication ipsec spi 500 md5 1234567890abcdef1234567890abcdef
```
auto-cost (OSPFv3)

To control how the Open Shortest Path First Version 3 (OSPFv3) protocol calculates default metrics for an interface, use the `auto-cost` command in an appropriate configuration mode. To set link cost based only on the interface type, use the `disable` form of this command. To re-enable OSPFv3 metric calculation for an interface according to the bandwidth of the interface, use the `no` form of this command.

```
auto-cost [{reference-bandwidth mbps|disable}]
no auto-cost [{reference-bandwidth mbps|disable}]
```

**Syntax Description**

- `reference-bandwidth mbps` (Optional) Sets the rate in Mbps (bandwidth). Range is 1 to 4294967.
- `disable` (Optional) Sets the link cost based only on the interface type.

**Command Default**

`mbps`: 100 Mbps

**Command Modes**

Router ospfv3 configuration

**Command History**

| Release 6.0 | This command was introduced. |

**Usage Guidelines**

By default OSPFv3 calculates the OSPFv3 metric for an interface according to the bandwidth of the interface. The `no auto-cost disable` form of this command reenables OSPFv3 metric calculation for an interface according to the bandwidth of the interface.

To set link cost based only on the interface type, use the `disable` keyword.

If you have multiple links with high bandwidth, you might want to use a larger number to differentiate the cost on those links.

Recommended usage of cost configuration for all OSPFv3 configured interfaces is to be consistent: Either explicitly configure link costs (by using the `cost` command) or choose an appropriate default (by using the `auto-cost` command).

The value set by the `cost` command overrides the cost resulting from the `auto-cost` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the reference value for the auto cost to 64:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# auto-cost reference-bandwidth 64
```
capability vrf-lite (OSPFv3)

To ignore DN bit in LSAs received from peers in the given VRF and to disable automatic ABR status in that VRF, use the capability vrf-lite command in OSPFv3 VRF configuration mode. To disable ignoring the DN bit in LSAs and to re-enable automatic ABR status in the VRF, use the no form of this command.

This feature is not supported.

capability vrf-lite
no capability vrf-lite

This command has no keywords or arguments.

Command Default
Disabled.

Command Modes
OSPFv3 VRF configuration

Command History
Release 6.0  This command was introduced.

Usage Guidelines
Use the capability vrf-lite command when routers (sometimes called multi-vrf CE routers) are directly connected through interfaces associated with a VRF, but without being connected to other PEs through the MPLS/VPN BGP Backbone.

When OSPFv3 is enabled in a VRF, the router is always an ABR. With the capability vrf-lite command enabled, the router becomes an ABR only if it is connected to area 0 (backbone area), and there are other (non-backbone) areas enabled on this router in the given VRF.

Routes may be re-introduced to the VPN backbone when this command is used.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to enable VRF-lite capability for OSPFv3 instance 1 under VRF vrf1:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)#vrf vrf1
RP/0/RP0/CPU0:router(config-ospfv3-vrf)#capability vrf-lite
```
clear ospfv3 process

To reset an Open Shortest Path First Version 3 (OSPFv3) router process without removing and reconfiguring it, use the `clear ospfv3 process` command in XR EXEC mode.

```
clear ospfv3 [process-name] process
```

**Syntax Description**

- **process-name** (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only the specified routing process is affected. Otherwise, all OSPFv3 processes are reset.

**Command Default**

No default behavior or value

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**

When the OSPFv3 router process is reset, OSPFv3 releases all resources allocated, cleans up the internal database, uninstalls routes, and resets all OSPFv3 adjacencies.

**Note**

The `clear ospfv3 process` command might clear the router ID configuration if the OSPF router ID is not explicitly configured through the `router-id (OSPFv3), on page 390` command.

**Examples**

The following example shows how to reset all OSPFv3 processes:

```
RP/0/RP0/CPU0:router# clear ospfv3 process
```

The following example shows how to reset the OSPFv3 process 1:

```
RP/0/RP0/CPU0:router# clear ospfv3 1 process
```
clear ospfv3 redistribution

To flush all the Type 5 and Type 7 link-state advertisements (LSAs) originated by an Open Shortest Path First Version 3 (OSPFv3) process, use the `clear ospfv3 redistribution` command in XR EXEC mode.

```
clear ospfv3 [process-name] redistribution
```

**Syntax Description**

- **process-name** (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only the specified routing process is affected. Otherwise, all OSPFv3 processes are reset.

**Command Default**

No default behavior or value

**Command Modes**

XR EXEC mode

**Command History**

```
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
```

**Usage Guidelines**

Use the `clear ospfv3 redistribution` command to cause the routing table to be read again. OSPFv3 regenerates and sends Type 5 and Type 7 link-state advertisements (LSAs) to its neighbors. If an unexpected route has appeared in the OSPFv3 redistribution, using this command corrects the issue.

**Note**

Use of this command can cause a significant number of LSAs to flood the network. We recommend that you use this command with caution.

**Task ID**

```
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
```

**Examples**

The following example shows how to clear all OSPFv3 redistributed routes from other protocols:

```
RP/0/RP0/CPU0:router# clear ospfv3 redistribution
```
clear ospfv3 routes

To clear the Open Shortest Path First Version 3 (OSPFv3) internal route table, use the `clear ospfv3 routes` command in XR EXEC mode.

```
clear ospfv3 [process-name] routes
```

**Syntax Description**

**process-name** (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only the specified routing process is affected. Otherwise, all OSPFv3 processes are reset.

**Command Default**

No default behavior or value

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `clear ospfv3 routes` command to force the internal route table to be repopulated by causing recalculation of the shortest path first (SPF) routing table. When the OSPFv3 routing table is cleared, OSPFv3 routes in the global routing table are also recalculated.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to clear all OSPFv3 routes from the OSPFv3 routing table and recomputes valid routes:

```
RP/0/RP0/CPU0:router# clear ospfv3 routes
```
clear ospfv3 statistics

To clear the Open Shortest Path First Version 3 (OSPFv3) statistical counters, use the `clear ospfv3 statistics` command in XR EXEC mode.

`clear ospfv3 [process-name] statistics [neighbor [type interface-path-id] [router-id]]`

**Syntax Description**

- **process-name**: (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only the specified routing process is affected.

- **neighbor**: (Optional) Clears counters for the specified neighbor only.

- **type**: Interface type. For more information, use the question mark (?) online help function.

- **interface-path-id**: Physical interface or virtual interface.

  **Note**: Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

  For more information about the syntax for the router, use the question mark (?) online help function.

- **router-id**: (Optional) Specified router ID. This argument must be in 32-bit dotted-decimal notation, similar to an IPv4 address. This argument clears the counters of the specified neighbor only.

**Command Default**

No default behavior or value

**Command Modes**

XR EXEC mode

**Command History**

- **Release**
  - 6.0

- **Modification**
  - This command was introduced.

**Usage Guidelines**

Use the `clear ospfv3 statistics` command to reset statistics so that subsequent changes are easily observed.

**Task ID**

- **Task ID**
  - ospf read, write

**Examples**

The following example shows how to clear the OSPFv3 statistical counters of all neighbors on tenGigE interface 0/2/0/0:

```
RP/0/RP0/CPU0:router# clear ospfv3 statistics neighbor tenGigE 0/2/0/0
```
cost (OSPFv3)

To explicitly specify the cost of the interface (network) for OSPF path calculations, use the `cost` command in an appropriate configuration mode. To remove the cost, use the `no` form of this command.

```
cost  cost
no  cost
```

**Syntax Description**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>cost</td>
</tr>
</tbody>
</table>

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the cost parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the cost parameter specified for the process.

If this command is not specified at any level, then the cost is based on the interface bandwidth, as specified by the `auto-cost` command.

**Command Modes**

- Interface configuration
- Area configuration
- Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The link-state metric is advertised as the link cost in the router link advertisement.

In general, the path cost is calculated using the following formula:

\[10^8 / \text{bandwidth}\]

The value set by the `cost` command overrides the cost resulting from the `auto-cost` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the cost value to 65 for tenGigE interface 0/1/0/1:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 201
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# cost 65
```
database-filter all out (OSPFv3)

To filter outgoing link-state advertisements (LSAs) to an Open Shortest Path First Version 3 (OSPFv3) interface, use the `database-filter all out` command in an appropriate configuration mode. To restore the forwarding of LSAs to the interface, use the `no` form of this command.

```
database-filter all out
no database-filter all out
```

**Syntax Description**

This command has no keywords or arguments.

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the database filter parameter specified for the area.

If this command is not specified in area configuration mode, then the interface adopts the database filter parameter specified for the process.

If this command is not specified in router ospfv3 configuration mode, then the database filter is disabled and all outgoing LSAs are flooded to the interface.

**Command Modes**

- Interface configuration
- Area configuration
- Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| Release 6.0 | This command was introduced.

**Usage Guidelines**

Use the `database-filter all out` command to perform the same function that the `neighbor` command (with the `database-filter` keyword) performs on a neighbor basis.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to prevent flooding of OSPFv3 LSAs to neighbors reachable through tenGigE interface 0/2/0/3:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/2/0/3
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# database-filter all out
```
dead-interval (OSPFv3)

To set the interval after which a neighbor is declared dead when no hello packets are observed, use the `dead-interval` command in an appropriate configuration mode. To return to the default time, use the `no` form of this command.

`dead-interval seconds`
`no dead-interval`

**Syntax Description**

| seconds | Unsigned integer that specifies the interval (in seconds). The value must be the same for all nodes on the same network link. Range is 1 to 65535. |

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the dead interval parameter specified for the area.

If this command is not specified in area configuration mode, then the interface adopts the dead interval parameter specified for the process.

If this command is not specified in router ospfv3 configuration mode, then the dead interval is four times the interval set by the `hello-interval (OSPFv3)` command.

**Command Modes**

- Interface configuration
- Area configuration
- Router OSPFv3 configuration
- Virtual-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Two Open Shortest Path First Version 3 (OSPFv3) routers do not become adjacent if their dead interval values differ.

If the hello interval is configured, the dead interval value must be larger than the hello interval value. The dead interval value is usually configured four times larger than the hello interval value.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the OSPFv3 dead interval on tenGigE interface 0/2/0/3 to 40 seconds:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
```
dead-interval (OSPFv3)

RP/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/2/0/3
RP/RP0/CPU0:router(config-ospfv3-ar-if)# dead-interval 40
**default-cost (OSPFv3)**

To specify a cost for the default summary route sent into a stub area or not-so-stubby area (NSSA) for Open Shortest Path First Version 3 (OSPFv3) packets, use the `default-cost` command in area configuration mode. To remove the assigned default route cost, use the `no` form of this command.

```
default-cost cost
no default-cost
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>cost</code></td>
<td>Cost for the default summary route used for a stub or NSSA area. The acceptable value is a 24-bit number ranging from 1 to 16777214.</td>
</tr>
</tbody>
</table>

**Command Default**

```
cost : 1
```

**Command Modes**

Area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `default-cost` command only on an Area Border Router (ABR) attached to a stub area or an NSSA area. In all routers and access servers attached to the stub area, the area should be configured as a stub area using the `stub (OSPFv3)` command in the area configuration submode. Use the `default-cost` command only on an ABR attached to the stub area. The `default-cost` command provides the metric for the summary default route generated by the ABR into the stub area.

**Examples**

The following example shows how to assign a cost of 20 to the default route sent into area 10.15.0.0:

```
RP/0/RP0/CPU0:router(config) # router ospfv3 201
RP/0/RP0/CPU0:router(config-ospfv3) # area 10.15.0.0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# stub
RP/0/RP0/CPU0:router(config-ospfv3-ar)# default-cost 20
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/3/0/1
```
default-information originate (OSPFv3)

To generate a default external route into an Open Shortest Path First Version 3 (OSPFv3) routing domain, use the `default-information originate` command in router ospfv3 configuration mode. To disable this feature, use the `no` form of this command.

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>route-policy</code></td>
<td>(Optional) Specifies the route policy to apply to default information origination.</td>
</tr>
<tr>
<td><code>policy-name</code></td>
<td>(Optional) Specifies the route policy to apply to default information origination.</td>
</tr>
<tr>
<td><code>always</code></td>
<td>(Optional) Always advertises the default route regardless of whether the software has a default route.</td>
</tr>
<tr>
<td><code>metric metric-value</code></td>
<td>(Optional) Specifies a metric used for generating the default route. The default metric value is 1. The value used is specific to the protocol.</td>
</tr>
<tr>
<td><code>metric-type</code></td>
<td>(Optional) Specifies an external link type associated with the default route advertised into the OSPFv3 routing domain. It can be one of the following values:</td>
</tr>
</tbody>
</table>
| `type-value`      | 1 — Type 1 external route  
|                   | 2 — Type 2 external route  
| `tag`             | (Optional) 32-bit dotted-decimal value attached to each external route. This is not used by the OSPFv3 protocol itself. It may be used to communicate information between autonomous system boundary routers (ASBRs). If a tag is not specified, then zero (0) is used. |

**Command Default**

A default external route into an OSPFv3 routing domain is not generated.

- **metric-value**: 1
- **type-value**: Type 2
- **tag-value**: 0

**Command Modes**

Router ospfv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Whenever you use the `redistribute` or `default-information` command to redistribute routes into an OSPFv3 routing domain, the software automatically becomes an ASBR. However, an ASBR does not, by default, generate a default route into the OSPFv3 routing domain. The software still must have a default route for itself before it generates one, except when you have specified the `always` keyword.
The `default-information originate` route-policy attach point conditionally injects the default route 0.0.0.0/0 into the OSPF link-state database, and is done by evaluating the attached policy. If any routes specified in the policy exist in the global RIB, then the default route is inserted into the link-state database. If there is no match condition specified in the policy, the policy passes and the default route is generated into the link-state database.

For information about the default-information originate attach point, see the `OSPF v3 Policy Attach Points` section in the `Implementing Routing Policy` chapter in *Routing Configuration Guide for Cisco NCS 5000 Series Routers*.

When you use the `default-information originate` command for the OSPFv3 process, the default network must reside in the routing table.

For information about routing policies, see the *Routing Policy Commands* chapter in the *Routing Command Reference for Cisco NCS 5000 Series Routers*.

### Examples

The following example shows how to specify a metric of 100 for the default route redistributed into the OSPFv3 routing domain and an external metric type of Type 1:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 109
RP/0/RP0/CPU0:router(config-ospfv3)# default-information originate metric 100 metric-type 1
```
default-metric (OSPFv3)

To set default metric values for routes redistributed from another protocol into Open Shortest Path First Version 3 (OSPFv3), use the `default-metric` command in an appropriate configuration mode. To return to the default state, use the `no` form of this command.

```
default-metric value
no default-metric value
```

**Syntax Description**

- `value` Default metric value appropriate for the specified routing protocol.

**Command Default**

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

**Command Modes**

Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `default-metric` command with the `redistribute` command to cause the current routing protocol to use the same metric value for all redistributed routes. A default metric helps solve the problem of redistributing routes with incompatible metrics. Whenever metrics do not convert, use a default metric to provide a reasonable substitute and enable the redistribution to proceed.

The default-metric value configured in OSPF configuration does not apply to connected routes that are redistributed to OSPF using the `redistribute connected` command. To set a non-default metric for connected routes, configure OSPF with the `redistribute connected metric metric-value` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure a router with both the Intermediate System-to-Intermediate System (IS-IS) and the OSPFv3 routing protocols. The OSPFv3 routing protocol advertises IS-IS derived routes and assigns the routes a metric of 10:

RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# default-metric 10
RP/0/RP0/CPU0:router(config-ospfv3)# redistribute isis IS-IS_isp
**demand-circuit (OSPFv3)**

To configure the Open Shortest Path First Version 3 (OSPFv3) router process to treat the interface as an OSPFv3 demand circuit, use the `demand-circuit` command in an appropriate configuration mode. To remove the demand circuit designation from the interface, use the `no` form of this command.

```
demand-circuit  [disable]
no  demand-circuit
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>disable  (Optional) Disables the demand circuit configuration that may have been specified at a higher level in the configuration.</th>
</tr>
</thead>
</table>

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the demand circuit parameter specified for the area.

If this command is not specified in area configuration mode, then the interface adopts the demand circuit parameter specified for the process.

If this command is not specified at any level, then the interface is not a demand circuit.

**Command Modes**

- Interface configuration
- Area configuration
- Router OSPFv3 configuration
- Virtual-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

On point-to-point interfaces, only one end of the demand circuit must be configured with the `demand-circuit` command. Periodic hello messages are suppressed and periodic refreshes of link-state advertisements (LSAs) do not flood the demand circuit. This command allows the underlying data link layer to be closed when the topology is stable. In point-to-multipoint topology, only the multipoint end must be configured with this command.

**Task ID**

- `ospf` read, write

**Examples**

The following example shows how to configure tenGigE interface 0/3/0/1 as an on-demand circuit:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3)# interface tenGigE 0/3/0/1
```
demand-circuit (OSPFv3)

RP/0/RP0/CPU0:router(config-ospfv3-if)# demand-circuit
**distance ospfv3**

To define the Open Shortest Path First Version 3 (OSPFv3) route administrative distances based on route type, use the `distance ospfv3` command in router ospfv3 configuration mode. To restore the default value, use the `no` form of this command.

```
distance ospfv3 {intra-area|inter-area|external} distance
no distance ospfv3
```

**Syntax Description**

- `intra-area` | `inter-area` | `external`  
  - Type of area. It can be one of the following values:
  - `intra-area` — All routes within an area.
  - `inter-area` — All routes from one area to another area.
  - `external` — All routes from other routing domains, learned by redistribution.

- `distance`  
  - The route administrative distance.

**Command Default**

`distance : 110`

**Command Modes**

Router ospfv3 configuration

---

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You must specify one of the keywords.

Use the `distance ospfv3` command to perform the same function as the `distance` command used with an access list. However, the `distance ospfv3` command sets a distance for an entire group of routes, rather than a specific route that passes an access list.

A common reason to use the `distance ospfv3` command is when you have multiple OSPFv3 processes with mutual redistribution, and you want to prefer internal routes from one over external routes from the other.

**Task ID**

- `ospf` read, write

**Examples**

The following example shows how to change the external distance to 200, making it less reliable:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# redistribute ospfv3 2
RP/0/RP0/CPU0:router(config-ospfv3)# distance ospfv3 external 200
RP/0/RP0/CPU0:router(config-ospfv3)# exit
RP/0/RP0/CPU0:router(config)# router ospfv3 2
RP/0/RP0/CPU0:router(config-ospfv3)# redistribute ospfv3 1
```
RP/0/RP0/CPU0:router(config-ospfv3)# distance ospfv3 external 200
distribute-list prefix-list in

To filter the routes that Open Shortest Path First Version 3 (OSPFv3) installs in the Routing Information Base (RIB), use the `distribute-list prefix-list in` command in an appropriate configuration mode. To remove the filter, use the `no` form of this command.

```
distribute-list prefix-list prefix-list-name in
no distribute-list prefix-list prefix-list-name in
```

**Syntax Description**

- **prefix-list-name**: IP Version 6 (IPv6) prefix list name. The list defines which IPv6 prefixes are installed in the RIB.

**Command Default**

All routes learned by OSPFv3 are installed in the RIB.

**Command Modes**

- Interface configuration
- Router OSPFv3 configuration

**Usage Guidelines**

Use the `distribute-list prefix-list` command to limit the routes that OSPFv3 installs in the RIB of your router. This command does not affect the information sent to other OSPFv3 routers or the routes that these routers compute and install.

**Note**

Because the other OSPFv3 routers are not aware of any omissions in the RIB, they may send traffic addressed to the missing prefixes. If no other provision has been made for these prefixes, the packets are dropped.

When this command is specified in router ospfv3 configuration mode, the filter applies to all routes computed by OSPFv3.

When this command is specified in interface configuration mode, the filter applies only to routes that forward outgoing traffic over that interface.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to prevent OSPFv3 from installing any routes that have 2001:e624 as the first 32 bits of the address. OSPFv3 is also prevented from installing routes to 2002::/16 that use tenGigE interface 0/2/0/0 as the next-hop interface:

```
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list preflist1
RP/0/RP0/CPU0:router(config-ipv6-pfx)# deny 2001:e624::/32 le 128
RP/0/RP0/CPU0:router(config-ipv6-pfx)# permit ::/0 le 128
!
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list preflist2
RP/0/RP0/CPU0:router(config-ipv6-pfx)# deny 2002::/16
RP/0/RP0/CPU0:router(config-ipv6-pfx)# permit ::/0 le 128
```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# distribute-list prefix-list preflist1 in
RP/0/RP0/CPU0:router(config-ospfv3)# area 1
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/2/0/0
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# distribute-list prefix-list preflist2 in
To filter the routes redistributed into Open Shortest Path First Version 3 (OSPFv3) from other routing protocols, use the `distribute-list prefix-list out` command in an appropriate configuration mode. To remove the filter, use the `no` form of this command.

```
distribute-list prefix-list prefix-list-name out [protocol [process-id]]
no distribute-list prefix-list prefix-list-name out [protocol [process-id]]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>prefix-list-name</td>
<td>IP Version 6 (IPv6) prefix list name. The list defines which IPv6 prefixes are installed in the RIB.</td>
</tr>
<tr>
<td>protocol</td>
<td>(Optional) Source protocol from which routes are being redistributed. It can be one of the following keywords: <code>bgp</code>, <code>isis</code>, <code>ospfv3</code>, <code>static</code>, and <code>connected</code>. The <code>static</code> keyword is used to redistribute IPv6 static routes. The <code>connected</code> keyword refers to routes that are established automatically because IPv6 is enabled on an interface. For routing protocols such as OSPFv3 and Intermediate System-to-Intermediate System (IS-IS), these routes are redistributed as external to the autonomous system.</td>
</tr>
</tbody>
</table>
| process-id | (Optional) For the `bgp` keyword, an autonomous system number has the following ranges:  
  - Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.  
  - Range for 4-byte Autonomous system numbers (ASNs) in asplain format is 1 to 4294967295.  
  - Range for 4-byte Autonomous system numbers (ASNs) as asdot format is 1.0 to 65535.65535.  
  For the `isis` keyword, an optional argument that defines a meaningful name for a routing process. You can specify only one IS-IS process for each router. Creating a name for a routing process means that you use names when configuring routing.  
  For the `ospfv3` keyword, an appropriate OSPFv3 process name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string. |

**Command Default**

All routes from protocols specified in the `redistribute (OSPFv3), on page 384` command are redistributed into OSPFv3.

**Command Modes**

Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Routes may be redistributed into OSPFv3 from several other routing protocols or from other OSPFv3 processes. These routes are then communicated to other OSPFv3 routes through Type 5 (External) or Type 7 not-so-stubby area (NSSA) link-state advertisements (LSAs). Use the `distribute-list prefix-list out` command to control...
redistribution by matching redistributed routes against an IPv6 prefix list. Only routes permitted by the prefix list are redistributed into OSPFv3.

Each protocol being redistributed into OSPFv3 can have a separate prefix list. In addition, a prefix list can be defined that applies to all protocols.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples

The following example shows how to prevent OSPFv3 from redistributing routes that have 2001:e624 as the first 32 bits of the address. In addition, routes with a prefix beginning with 2064 are not redistributed from Border Gateway Protocol (BGP) autonomous system 1, and only those routes are redistributed from BGP autonomous system 5.

```
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list p1
RP/0/RP0/CPU0:router(config-ipv6-pfx)# deny 2001:e624::/32 le 128
RP/0/RP0/CPU0:router(config-ipv6-pfx)# permit ::/0 le 128
!
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list p2
RP/0/RP0/CPU0:router(config-ipv6-pfx)# deny 2064::/16 le 128
RP/0/RP0/CPU0:router(config-ipv6-pfx)# permit ::/0 le 128
!
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list p3
RP/0/RP0/CPU0:router(config-ipv6-pfx)# permit 2064::/16 le 128
!
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# redistribute bgp 1
RP/0/RP0/CPU0:router(config-ospfv3)# redistribute bgp 5
RP/0/RP0/CPU0:router(config-ospfv3)# distribute-list prefix-list p1 out
RP/0/RP0/CPU0:router(config-ospfv3)# distribute-list prefix-list p2 out bgp 1
RP/0/RP0/CPU0:router(config-ospfv3)# distribute-list prefix-list p3 out bgp 5
```
domain-id (OSPFv3)

To specify the Open Shortest Path First Version 3 (OSPFv3) VPN routing and forwarding (VRF) domain ID, use the `domain-id` command in VRF configuration mode. To remove the OSPFv3 VRF domain ID, use the `no` form of this command.

```
domain-id [secondary] type [{0005|0105|0205}] value domain-id_value
no domain-id [secondary] type [{0005|0105|0205}] value domain-id-value
```

**Syntax Description**

<table>
<thead>
<tr>
<th>secondary</th>
<th>(Optional) OSPFv3 secondary domain ID.</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Primary OSPFv3 domain ID in hexadecimal format.</td>
</tr>
<tr>
<td></td>
<td>• 0005 — Type 0x0005</td>
</tr>
<tr>
<td></td>
<td>• 0105 — Type 0x0105</td>
</tr>
<tr>
<td></td>
<td>• 0205 — Type 0x0205</td>
</tr>
<tr>
<td>value</td>
<td>OSPF domain ID value in hexadecimal format.</td>
</tr>
<tr>
<td>domain-id-value</td>
<td>OSPF domain ID extended community value as a 6 byte hexadecimal number.</td>
</tr>
</tbody>
</table>

**Command Default**

No domain ID is specified.

**Command Modes**

VRF configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If no value is specified for the domain ID, the default is Null (all zeros) primary domain ID. One or more secondary domain IDs can be specified.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to specify a domain ID with type 0105 and value AABBCDDEEFF:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf 1
RP/0/RP0/CPU0:router(config-ospf)# vrf vrf_1
RP/0/RP0/CPU0:router(config-ospf-vrf)# domain-id type 0105 value AABBCDDEEFF
```
To encrypt and authenticate Open Shortest Path First Version 3 (OSPFv3) packets, use the `encryption` command in an appropriate configuration mode. To remove the encryption, use the `no` form of this command.

```
encryption {disable|ipsec spi spi-value esp {3des|aes [{192|256}]|des|null [{clear|password}]}}
encrypt-password} [authentication {md5|sha1} [{clear|password}] auth-password]}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disable</td>
<td>Disables OSPFv3 packet encryption.</td>
</tr>
<tr>
<td>ipsec spi</td>
<td>Specifies IPSec ESP encryption and authentication with the Security Parameter Index (SPI) value.</td>
</tr>
<tr>
<td>spi-value</td>
<td>SPI value. Range is 256 to 4294967295.</td>
</tr>
<tr>
<td>esp</td>
<td>Specifies Encryption Service Payload (ESP) encryption parameters.</td>
</tr>
<tr>
<td>3des</td>
<td>Specifies the triple DES algorithm.</td>
</tr>
<tr>
<td>aes</td>
<td>Specifies the Advanced Encryption Standard (AES) algorithm.</td>
</tr>
<tr>
<td>192</td>
<td>(Optional) Specifies the 192-bit AES algorithm.</td>
</tr>
<tr>
<td>256</td>
<td>(Optional) Specifies the 256-bit AES algorithm.</td>
</tr>
<tr>
<td>des</td>
<td>Specifies the Data Encryption Standard (DES) algorithm.</td>
</tr>
<tr>
<td>null</td>
<td>Specifies no AES algorithm.</td>
</tr>
<tr>
<td>md5</td>
<td>Enables Message Digest 5 (MD5) authentication.</td>
</tr>
<tr>
<td>sha1</td>
<td>Enables SHA1 authentication.</td>
</tr>
<tr>
<td>clear</td>
<td>Specifies that the key be unencrypted.</td>
</tr>
<tr>
<td>password</td>
<td>Specifies that the key be encrypted using a two-way algorithm.</td>
</tr>
<tr>
<td>encrypt-password</td>
<td>Any contiguous string that can be entered from the keyboard as the encryption password.</td>
</tr>
<tr>
<td>auth-password</td>
<td>Any contiguous string that can be entered from the keyboard as the authentication password.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values.

**Command Modes**

Interface configuration

Router OSPFv3 configuration
**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **encryption** command to encrypt and authenticate OSPFv3 packets.

**Note**

- IPSec is supported only for Open Shortest Path First version 3 (OSPFv3).
- If OSPFv3 is configured along with IPsec authentication, then it is likely that adjacencies may flap on a Route Processor Fail Over (RPFO) even when NSR and/or Graceful Restart is enabled.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to encrypt and authenticate OSPFv3 packets:

```
RP/0/RP0/CPU0:router(config) router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)#encryption ipsec spi 256 esp 3des clear
```
flood-reduction (OSPFv3)

To suppress the unnecessary flooding of link-state advertisements (LSAs) in stable topologies, use the flood-reduction command in an appropriate configuration mode. To disable this feature, use the no form of this command.

flood-reduction [disable]
no flood-reduction

Syntax Description
- **disable** (Optional) Turns off this functionality at a specific level.
- **Note** The disable keyword is not available in router ospfv3 configuration mode.

Command Default
- If this command is not specified in interface configuration mode, then the interface adopts the flood reduction parameter specified by area.
- If this command is not specified in area configuration mode, then the interface adopts the flood reduction parameter specified for the process.
- If this command is not specified at any level, then flood reduction is disabled.

Command Modes
- Interface configuration
- Area configuration
- Router OSPFv3 configuration

Command History
- **Release** **Modification**
  - Release 6.0 This command was introduced.

Usage Guidelines
- All routers that support Open Shortest Path First Version 3 (OSPFv3) demand circuits are compatible with routers supporting flooding reduction.

Task ID
- **Task ID**
  - **Operations**
    - ospf read, write

Examples
The following example shows how to reduce the flooding of unnecessary LSAs for area 0:

RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# flood-reduction
graceful-restart (OSPFv3)

To enable graceful restart, use the `graceful-restart` command in an appropriate configuration mode. To disable this feature, use the `no` form of this command.

```
graceful-restart [{helper disable|interval interval lifetime lifetime}]  
no graceful-restart [{helper disable|interval interval lifetime lifetime}]  
```

### Syntax Description

- **helper disable** (Optional) Disables the routers helper support level.
- **interval interval** (Optional) Specifies the minimum interval between graceful restarts. Range is 90 to 3600 seconds.
- **lifetime lifetime** (Optional) Specifies the maximum route lifetime following a restart. Range is 90 to 3600 seconds.

### Command Default

No default behavior or values.

### Command Modes

Router OSPFv3 configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

No specific guidelines impact the use of this command.

### Examples

The following example shows how to enable the Graceful Restart feature with a minimum interval between restarts of 300 seconds:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# graceful-restart interval 300
```
hello-interval (OSPFv3)

To specify the interval between hello packets that Open Shortest Path First Version 3 (OSPFv3) sends on an interface, use the **hello-interval** command in an appropriate configuration mode. To return to the default time, use the **no** form of this command.

**hello-interval** *seconds*
**no** hello-interval

**Syntax Description**

*seconds* Interval (in seconds). The value must be the same for all nodes on a specific network.

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the hello interval parameter specified by area.

If this command is not specified in area configuration mode, then the interface adopts the hello interval parameter specified for the process.

If this command is not specified at any level, then the hello interval is 10 seconds (broadcast) or 30 seconds (non-broadcast).

**Command Modes**

Interface configuration
Area configuration
Router OSPFv3 configuration
Virtual-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The hello interval value is advertised in the hello packets. The shorter the hello interval, the faster topological changes are detected, but more routing traffic ensues. This value must be the same for all routers and access servers on a specific network.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the interval between hello packets to 15 seconds on GigabitEthernet interface 0/3/0/2:

```
RP/0/RP0/CPU0:router (config)# router ospfv3 1
RP/0/RP0/CPU0:router (config-ospfv3)# area 0
RP/0/RP0/CPU0:router (config-ospfv3-ar)# interface GigabitEthernet 0/3/0/2
RP/0/RP0/CPU0:router (config-ospfv3-ar-if)# hello-interval 15
```
hello-interval (OSPFv3)
instance (OSPFv3)

To set the 8-bit instance ID used in Open Shortest Path First Version 3 (OSPFv3) packets sent on an interface, use the `instance` command in an appropriate configuration mode. To remove the instance ID, use the `no` form of this command.

```
instance instance-id
no instance instance-id
```

**Syntax Description**

- `instance-id` Instance identifier sent in OSPFv3 packets. Range is 0 to 255. The same value must be used by all the communicating OSPFv3 routers on a link.

**Command Default**

- If this command is not specified in interface configuration mode, then the interface adopts the instance parameter specified by the area.
- If this command is not specified in area configuration mode, then the interface adopts the instance parameter specified for the process.
- If this command is not specified at any level, then the instance is 0.

**Command Modes**

- Interface configuration
- Area configuration
- Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The OSPFv3 routing protocol allows multiple unrelated OSPFv3 processes to share a link by using an 8-bit “instance” value to demultiplex the protocol packets. Each OSPFv3 process sets its configured instance value in the OSPFv3 packets that it sends, and ignores received packets with instance values from other OSPFv3 processes.

> The **instance-id** argument should not be confused with the **process-name** argument that is specified by the `router ospfv3` command. The former is an 8-bit integer that is sent to other routers as part of the OSPFv3 protocol, and the latter is a 1- to 40-character ASCII string that is significant only within a given router. The instance ID value is also unrelated to the router ID that is specified by the `router-id` command, which is a 32-bit integer value that uniquely identifies a router within an OSPFv3 routing domain.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to set the instance value for tenGigE interface 0/3/0/1 to 42:

```bash
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/3/0/1
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# instance 42
```
interface (OSPFv3)

To define the interfaces on which Open Shortest Path First Version 3 (OSPFv3) runs, use the `interface` command in an appropriate configuration mode. To disable OSPFv3 routing for an interface, use the `no` form of this command.

```
interface type interface-path-id
no interface type interface-path-id
```

**Syntax Description**

- **type** Interface type. For more information, use the question mark (?) online help function.
- **interface-path-id** Physical interface or virtual interface.

**Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

**Command Default**

An interface is not defined.

**Command Modes**

Area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**

Use the `interface` command to associate a specific interface with an OSPFv3 area. The interface remains associated with the area even when the IPv6 address of the interface changes.

Similar to IPv4 address behavior for the `interface` command, all configured IPv6 addresses are advertised on an interface after the interface is associated to the OSPF routing process. The only difference is, IPv6 addresses can have multiple primary addresses.

This command places the router in interface configuration mode (prompt: config-router-ar-if), from which you can configure interface-specific settings. Commands configured under this mode (such as the `cost` command) are automatically bound to that interface.

**Examples**

The following example shows how to define two interfaces that belong to area 1. The cost value for packets on tenGigE interface 0/3/0/1 is set at 40; the cost value for tenGigE interface 0/3/0/2 is 65:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 1
```
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/3/0/1
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# cost 40
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# exit
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/3/0/2
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# cost 65
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# exit
log adjacency changes (OSPFv3)

To change the default syslog messages for Open Shortest Path First Version 3 (OSPFv3) neighbor state changes, use the log adjacency changes command in an appropriate configuration mode. To suppress all adjacency change messages, use the disable keyword.

```
log adjacency changes [{detail|disable}]
```

**Syntax Description**

- **detail** (Optional) Provides all (DOWN, INIT, 2WAY, EXSTART, EXCHANGE, LOADING, FULL) adjacency state changes.
- **disable** (Optional) Disables the neighbor state change messages.

**Command Default**

Neighbor state change messages are enabled.

**Command Modes**

Router OSPFv3 configuration

**Command History**

```
Release   Modification
----------  ------------------
Release 6.0  This command was introduced.
```

**Usage Guidelines**

By default, you are notified of OSPFv3 neighbor changes without explicitly configuring the log adjacency changes command. The syslog messages that are sent provide a high-level view of changes to the state of the OSPFv3 peer relationship.

**Task ID**

```
Task ID  Operations
---------  ------------
ospf      read, write
```

**Examples**

The following example shows how to disable neighbor state change messages:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# log adjacency changes disable
```

The following example shows how to re-enable syslog messages for any OSPFv3 neighbor state changes:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# log adjacency changes
```
**maximum interfaces (OSPFv3)**

To control the maximum number of interfaces that can be configured under an Open Shortest Path First Version 3 (OSPFv3) process, use the `maximum interfaces` command in an appropriate configuration mode. To remove the `maximum interfaces` command from the configuration file and restore the system to its default condition with respect to the routing protocol, use the `no` form of this command.

```
maximum interfaces number-interfaces
no maximum interfaces
```

**Syntax Description**

- `number-interfaces` Maximum number of interfaces that can be configured for this OSPFv3 process. Range is 1 to 1024. Range is 1 to 4294967295.

**Command Default**

- If the command is not specified, the default is 1024.

**Command Modes**

- Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

- No specific guidelines impact the use of this command.

**Task ID**

- `ospf` read, write

**Examples**

This example shows how to allow a maximum of 500 interfaces in an OSPFv3 process:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# maximum interfaces 500
```

This example shows how to allow a maximum of 1500 interfaces in an OSPFv3 process:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# maximum interfaces 1500
```
maximum paths (OSPFv3)

To control the maximum number of parallel routes that the Open Shortest Path First Version 3 (OSPFv3) can support, use the `maximum paths` command in an appropriate configuration mode. To remove the `maximum paths` command from the configuration file and restore the system to its default condition with respect to the routing protocol, use the `no` form of this command.

```
maximum paths maximum-routes-number
no maximum paths
```

**Syntax Description**

- `maximum-routes-number`: Maximum number of parallel routes that OSPFv3 can install in a routing table.
  - Range is 1 to 32.
  - **Note**: The maximum number of paths that can be configured is 32.

**Command Default**

32 paths

**Command Modes**

Router OSPFv3 configuration

**Command History**

- **Release 6.0**: This command was introduced.

**Usage Guidelines**

When the maximum number of parallel routes is reduced, all existing paths are pruned and paths reinstalled at the new maximum number. During this route-reduction period, you may experience some packet loss for a few seconds. This may impact route traffic.

**Task ID**

- `ospf` read, write

**Examples**

The following example shows how to allow a maximum of two paths to a destination:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# maximum paths 2
```
maximum redistributed-prefixes (OSPFv3)

To limit the number of prefixes redistributed into Open Shortest Path First Version 3 (OSPFv3) or to generate a warning when the number of prefixes redistributed into OSPFv3 reaches a maximum, use the maximum redistributed-prefixes command in an appropriate configuration mode. To remove the values, use the no form of this command.

maximum redistributed-prefixes limit [threshold] [warning-only]
no maximum redistributed-prefixes

Syntax Description

- **limit**: Maximum number of IP Version 6 (IPv6) prefixes that are allowed to be redistributed into OSPFv3, or, if the warning-only keyword is present, sets the number of prefixes allowed to be redistributed into OSPFv3 before the system logs a warning message. Range is 1 to 4294967295.

  **Note**: If the warning-only keyword is also configured, this value does not limit redistribution; it is simply the number of redistributed prefixes that, when reached, causes a warning message to be logged.

- **threshold** (Optional): Percentage of the value set for the maximum number of redistributed prefixes that, when reached, causes a warning message to be logged.

- **warning-only** (Optional): Causes a warning to be logged when the number of routes defined by the limit argument have been redistributed. Additional redistribution is not prevented.

Command Default

- **limit**: 10240
- **threshold**: 75 percent

Command Modes

- Router OSPFv3 configuration

Command History

- **Release 6.0**: This command was introduced.

Usage Guidelines

If someone mistakenly injects a large number of IPv6 routes into OSPFv3, perhaps by redistributing Border Gateway Protocol (BGP) into OSPFv3, the network can be severely flooded. Limiting the number of redistributed routes prevents this potential problem.

When the maximum redistributed-prefixes command is configured, if the number of redistributed routes reaches the maximum value configured, no more routes are redistributed (unless the warning-only keyword is configured).

The redistribution limit applies only to external IPv6 prefixes. Default routes and summarized routes are not limited.

The limit is tracked separately for each not-so-stubby-area (NSSA) because redistribution to NSSAs is done independently for each NSSA and independently of all other regular areas.

Select a maximum value based on your knowledge of how many prefixes are redistributed on the router to the OSPFv3 process.
This example shows how to set a maximum of 2000 prefixes that can be redistributed into OSPFv3 process 1. If the number of prefixes redistributed reaches 75 percent of 2000 (1500 prefixes), a warning message is logged. Another warning is logged if the limit is reached and no more routes are redistributed.

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# redistribute bgp 2406
RP/0/RP0/CPU0:router(config-ospfv3)# maximum redistributed-prefixes 2000
```
**mtu-ignore (OSPFv3)**

To prevent the Open Shortest Path First Version 3 (OSPFv3) router process from checking whether neighbors are using the same maximum transmission unit (MTU) on a common interface when exchanging database descriptor (DBD) packets, use the `mtu-ignore` command in an appropriate configuration mode. To return to the default state, use the `no` form of this command.

```
mtu-ignore [disable]
no mtu-ignore
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disable</td>
<td>(Optional) Disables the attribute in instances in which it is specified at a higher level in the configuration.</td>
</tr>
</tbody>
</table>

**Note**

The `disable` keyword is not available in router ospfv3 configuration mode.

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the MTU ignore parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the MTU ignore parameter specified for the process.

If this command is not specified at any level, then OSPFv3 checks the MTU received from neighbors when exchanging DBD packets.

**Command Modes**

- Interface configuration
- Area configuration
- Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `mtu-ignore` command to check whether OSPFv3 neighbors are using the same MTU on a common interface. This check is performed when neighbors exchange DBD packets. If the receiving MTU in the DBD packet is higher than the MTU configured on the incoming interface, OSPF adjacency is not established.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to disable MTU mismatch detection on received DBD packets on tenGigE interface 0/0/3:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
```
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# mtu-ignore
neighbor (OSPFv3)

To configure Open Shortest Path First Version 3 (OSPFv3) routers interconnecting to nonbroadcast networks, use the `neighbor` command in interface configuration mode. To remove a configuration, use the `no` form of this command.

```
neighbor ipv6-address [priority number] [poll-interval seconds] [cost number] [database-filter all out]
no neighbor ipv6-address [priority number] [poll-interval seconds] [cost number] [database-filter all out]
```

### Syntax Description

- **ipv6-address**
  - Link-local IP Version 6 (IPv6) address of the neighbor. This argument must be in the form documented in RFC 2373, in which the address is specified in hexadecimal using 16-bit values between colons.

- **priority number**
  - (Optional) Specifies an 8-bit number indicating the router priority value of the nonbroadcast neighbor associated with the IP address specified. The `priority` keyword does not apply to point-to-multipoint interfaces.

- **poll-interval seconds**
  - (Optional) Specifies an unsigned integer value (in seconds) reflecting the poll interval. RFC 1247 recommends that this value be much larger than the hello interval. The `poll-interval` keyword does not apply to point-to-multipoint interfaces.

- **cost number**
  - (Optional) Assigns a cost to the neighbor, in the form of an integer from 1 to 65535. Neighbors with no specific cost configured assume the cost of the interface, based on the `cost` command. On point-to-multipoint interfaces, `cost number` is the only keyword and argument combination that works. The `cost` keyword does not apply to nonbroadcast multiaccess (NBMA) networks.

- **database-filter all out**
  - (Optional) Filters outgoing link-state advertisements (LSAs) to an OSPFv3 neighbor.

### Command Default

- No configuration is specified.
- **priority number**: 0
- **poll-interval seconds**: 120 seconds (2 minutes)

### Command Modes

- Interface configuration

### Command History

- **Release 6.0** This command was introduced.

### Usage Guidelines

One neighbor entry must be included in the software configuration for each known nonbroadcast network neighbor. The neighbor address must be the IPv6 link-local address of the interface.
If a neighboring router has become inactive (hello packets have not been seen for the router dead interval period), it may still be necessary to send hello packets to the dead neighbor. These hello packets are sent at a reduced rate called the poll interval.

When the router starts up, it sends only hello packets to those routers with nonzero priority; that is, routers that are eligible to become designated routers (DRs) and backup designated routers (BDRs). After the DR and BDR are selected, the DR and BDR then start sending hello packets to all neighbors to form adjacencies.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to declare a router at address fe80::3203:a0ff:fe9d:f3fe on a nonbroadcast network:

```bash
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/2/0/0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# network non-broadcast
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# neighbor fe80::3203:a0ff:fe9d:f3fe
```
network (OSPFv3)

To configure the Open Shortest Path First Version 3 (OSPFv3) network type to a type other than the default for a given medium, use the `network` command in an appropriate configuration mode. To return to the default value, use the `no` form of this command.

```
network  {broadcast|non-broadcast} [point-to-multipoint]  
no  network
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>broadcast</td>
<td>Sets the network type to broadcast.</td>
</tr>
<tr>
<td>non-broadcast</td>
<td>Sets the network type to nonbroadcast multiaccess (NBMA).</td>
</tr>
<tr>
<td>point-to-multipoint</td>
<td>Sets the network type to point-to-multipoint.</td>
</tr>
</tbody>
</table>

**[non-broadcast]**

(Optional) Sets the point-to-multipoint network to be nonbroadcast. If you use the `non-broadcast` keyword, the `neighbor` command is required.

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>point-to-point</td>
<td>Sets the network type to point-to-point.</td>
</tr>
</tbody>
</table>

### Command Default

If this command is not specified in interface configuration mode, then the interface adopts the network parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the network parameter specified for the process.

If this command is not specified at any level, then the OSPFv3 network type is the default of the given medium.

### Command Modes

<table>
<thead>
<tr>
<th>Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface configuration</td>
<td></td>
</tr>
<tr>
<td>Area configuration</td>
<td></td>
</tr>
<tr>
<td>Router OSPFv3 configuration</td>
<td></td>
</tr>
</tbody>
</table>

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `network` command to configure broadcast networks as NBMA networks when, for example, routers in your network do not support multicast addressing.

Most times, it is assumed that when you configure NBMA networks as either broadcast or nonbroadcast, there are virtual circuits from every router to every router or fully meshed network. However, there are other configurations where this assumption is not true; for example, a partially meshed network. In these cases, you can configure the OSPFv3 network type as a point-to-multipoint network. Routing between two routers that are not directly connected goes through the router that has virtual circuits to both routers. You need not configure neighbors when using this command.

If the `network` command is issued on an interface that does not allow it, this command is ignored.

OSPFv3 has two features related to point-to-multipoint networks. One feature applies to broadcast networks and the other feature applies to nonbroadcast networks:
• On point-to-multipoint, broadcast networks, you can use the `neighbor` command, and you must specify a cost to that neighbor.

• On point-to-multipoint, nonbroadcast networks, you must use the `neighbor` command to identify neighbors. Assigning a cost to a neighbor is optional.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure an Ethernet interface as point-to-point:

```
RP/0/RP0/CPU0:router (config)# router ospfv3 1
RP/0/RP0/CPU0:router (config-ospfv3)# area 0
RP/0/RP0/CPU0:router (config-ospfv3-ar)# interface tenGigE 0/1/0/3
RP/0/RP0/CPU0:router (config-ospfv3-ar-if)# network point-to-point
```
**nssa (OSPFv3)**

To configure an area as a not-so-stubby area (NSSA), use the `nssa` command in area configuration mode. To remove the NSSA distinction from the area, use the `no` form of this command.

```
nssa [no-redistribution] [default-information-originate [{metric metric-value|metric-type type-value}]]
[no-summary]
```

**Syntax Description**

- **no-redistribution** (Optional) Imports routes into the normal areas, but not into the NSSA area, by the `redistribute` command when the router is an NSSA area border router (ABR).

- **default-information-originate** (Optional) Generates a Type 7 default into the NSSA area. This keyword takes effect only on an NSSA ABR or NSSA autonomous system boundary router (ASBR).

- **metric metric-value** (Optional) Specifies a metric used for generating the default route. If you do not specify a default route metric value using the `nssa` and `default-metric` commands, the default metric value is 10. The value used is specific to the protocol.

- **metric-type type-value** (Optional) Specifies an external link type associated with the default route advertised into the Open Shortest Path First Version 3 (OSPFv3) routing domain. It can be one of the following values:
  - 1 — Type 1 external route
  - 2 — Type 2 external route

- **no-summary** (Optional) Prevents an (ABR) from sending summary link advertisements into the NSSA area.

**Command Default**

No NSSA area is defined.

If you do not specify a value using the `default-metric` command, the default metric value is 10.

The default `type-value` is Type 2 external route.

**Command Modes**

Area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

A default route need not be defined in an NSSA ABR when the `nssa` command is configured. However, if this command is configured on an NSSA ASBR, then a default route must be defined.

**Note**

NSSA cannot be configured for area 0 (backbone area).
### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to configure area 1 as an NSSA area:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# router-id 10.18.1.1
RP/0/RP0/CPU0:router(config-ospfv3)# area 1
RP/0/RP0/CPU0:router(config-ospfv3-ar)# nssa
```
ospfv3 name-lookup

To configure Open Shortest Path First Version 3 (OSPFv3) to look up Domain Name System (DNS) names, use the `ospfv3 name-lookup` command in XR Config mode mode. To disable this function, use the `no` form of this command.

```
ospfv3 name-lookup
no ospfv3 name-lookup
```

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

Routers are displayed by router ID or neighbor ID.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `ospfv3 name-lookup` command to simplify the task of searching for a router. Routers are displayed by name rather than by router ID or neighbor ID.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure OSPFv3 to look up DNS names for use in all OSPFv3 `show` command displays:

```
RP/0/RP0/CPU0:router(config)# ospfv3 name-lookup
```
packet-size (OSPFv3)

To configure the size of Open Shortest Path First Version 3 (OSPFv3) packets up to the size specified by the maximum transmission unit (MTU), use the `packet-size` command in an appropriate configuration mode. To disable this function and reestablish the default packet size, use the `no` form of this command.

```
packet-size bytes
no packet-size
```

**Syntax Description**
- `bytes` Size in bytes. Range is 256 to 10000 bytes.

**Command Default**
If not specified, the default packet size is 1500 bytes.

**Command Modes**
- Router OSPFv3 configuration
- Area configuration
- Interface configuration

**Command History**

```
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
```

**Usage Guidelines**
Use the `packet-size` command to customize the size of OSPFv3 packets. The OSPFv3 protocol compares the packet size and the MTU size and uses the lower packet size value.

**Task ID**
```
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
```

**Examples**
The following example shows how to configure the packet size:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospf osp3
RP/0/RP0/CPU0:router(config-ospf3)# packet-size 3500
```
passive (OSPFv3)

To suppress the sending of Open Shortest Path First Version 3 (OSPFv3) packets on an interface, use the `passive` command in an appropriate configuration mode. To remove the passive configuration, use the `no` form of this command.

```
passive [disable]
no passive
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>disable (Optional) Sends OSPFv3 updates.</td>
<td></td>
</tr>
</tbody>
</table>

**Note** The `disable` keyword is not available in router ospfv3 configuration mode.

### Command Default

If this command is not specified in interface configuration mode, then the interface adopts the passive parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the passive parameter specified for the process.

If this command is not specified at any level, then the passive parameter is disabled and OSPFv3 updates are sent on the interface.

### Command Modes

- Interface configuration
- Area configuration
- Router OSPFv3 configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

OSPF routing information is neither sent nor received through the specified router interface. The specified interface address appears as a stub network in the OSPF domain.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows that OSPFv3 updates run over tenGigE interface 0/3/0, 0/2/0/0, and 0/2/0/2. All other interfaces suppress sending OSPFv3 updates because they are in passive mode.

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# router-id 10.0.0.206
RP/0/RP0/CPU0:router(config-ospfv3)# passive
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
```
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# passive disable
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# exit
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/3/0/1
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# exit
RP/0/RP0/CPU0:router(config-ospfv3-ar)# exit
RP/0/RP0/CPU0:router(config-ospfv3-ar)# area 1
RP/0/RP0/CPU0:router(config-ospfv3-ar)# passive disable
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/2/0/0
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# exit
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# passive
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# exit
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/2/0/2
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# exit
priority (OSPFv3)

To set the router priority for an interface, which helps determine the designated router for an Open Shortest Path First Version 3 (OSPFv3) link, use the `priority` command in an appropriate configuration mode. To return to the default value, use the `no` form of this command.

**Syntax Description**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>value</code></td>
<td>8-bit unsigned integer indicating the router priority value. Range is 0 to 255.</td>
</tr>
</tbody>
</table>

**Command Default**

If this command is not specified in interface configuration mode, then the interface adopts the priority parameter specified by the area.

If this command is not specified in area configuration mode, then the interface adopts the priority parameter specified by the process.

If this command is not specified at any level, then the default priority is 1.

**Command Modes**

- Interface configuration
- Area configuration
- Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

When two routers attached to a network both attempt to become the designated router, the one with the higher router priority takes precedence. If there is a tie, the router with the higher router ID takes precedence. A router with a router priority set to zero is ineligible to become the designated router or backup designated router. Router priority is configured only for interfaces to broadcast and nonbroadcast multiaccess (NBMA) networks.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the router priority value to 4 on tenGigE interface 0/1/0/1:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# priority 4
```
**protocol shutdown (OSPFv3)**

To disable an instance of the Open Shortest Path First protocol, version 3 (OSPFv3), so that it cannot form an adjacency on any interface, use the `protocol shutdown` command in the ospfv3 configuration mode. To re-enable the OSPF protocol, use the `no` form of this command.

```
protocol shutdown
no protocol shutdown
```

**Command Default**

None

**Command Modes**

ospfv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `protocol shutdown` command to disable the OSPFv3 protocol for a specific routing instance without removing any existing OSPF configuration parameters.

The OSPFv3 protocol continues to run on the router and you can use the current OSPFv3 configuration, but OSPFv3 does not form any adjacencies on any interface.

This command is similar to performing the `no router ospf` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

This example shows how to disable the OSPFv3:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospfv3 100
RP/0/RP0/CPU0:router(config-ospfv3)# protocol shutdown
```
range (OSPFv3)

To consolidate and summarize routes at an area boundary for Open Shortest Path First Version 3 (OSPFv3), use the range command in area configuration mode. To restore the default values, use the no form of this command.

```
range ipv6-prefix/prefix-length [{advertise|not-advertise}] [cost number]
no range ipv6-prefix/prefix-length [{advertise|not-advertise}] [cost number]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv6-prefix</td>
<td>Summary prefix designated for a range of IP Version 6 (IPv6) prefixes.</td>
</tr>
<tr>
<td>/ prefix-length</td>
<td>Length of the IPv6 prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash must precede the decimal value.</td>
</tr>
<tr>
<td>advertise</td>
<td>(Optional) Sets the address range status to advertise and generates a Type 3 summary link-state advertisement (LSA).</td>
</tr>
<tr>
<td>not-advertise</td>
<td>(Optional) Sets the address range status to DoNotAdvertise. The Type 3 summary LSA is suppressed and the component networks remain hidden from other networks.</td>
</tr>
<tr>
<td>cost number</td>
<td>(Optional) Specifies a cost for the range. Range is 1 to 16777214.</td>
</tr>
</tbody>
</table>

### Command Default

Routes are not consolidated and summarized for an area.

### Command Modes

Area configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the range command only with Area Border Routers (ABRs). It is used to consolidate or summarize routes for an area. The result is that a single summary route is advertised to other areas by the ABR. Routing information is condensed at area boundaries. External to the area, a single route is advertised for each address range. This process is called route summarization.

You can use the range command to configure multiple ranges. Thus, OSPFv3 can summarize addresses for many different sets of address ranges.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to specify one summary route to be advertised by the ABR to other areas for all IPv6 prefixes within the range defined by summary prefix 4004:f000::/32:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 201
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# range 4004:f000::/32
```
redistribute (OSPFv3)

To redistribute routes from one routing domain into Open Shortest Path First Version 3 (OSPFv3), use the redistribute command in an appropriate configuration mode. To remove the redistribute command from the configuration file and restore the system to its default condition in which the software does not redistribute routes, use the no form of this command.

**Border Gateway Protocol (BGP)**

redistribute bgp process-id [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

no redistribute bgp process-id [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

**Local Interface Routes**

redistribute connected [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

no redistribute connected [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

**Intermediate System-to-Intermediate System (IS-IS)**

redistribute isis process-id [{level-1|level-2|level-1-2}] [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

no redistribute isis process-id [{level-1|level-2|level-1-2}] [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

**Open Shortest Path First Version 3 (OSPFv3)**

redistribute ospfv3 process-id match[{external|1|2|internal|nssa-external|nssa-external}|{1|2}] [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

no redistribute ospfv3 process-id [{match|external|internal|nssa-external}] [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

**Static**

redistribute static [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

no redistribute static [metric metric-value] [metric-type {1|2}] [policy policy-name] [tag tag-value]

**Syntax Description**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgp</td>
<td>Distributes routes from the BGP protocol.</td>
</tr>
</tbody>
</table>
Process Id

For the `bgp` keyword, an autonomous system number has the following ranges:

- Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.
- Range for 4-byte Autonomous system numbers (ASNs) in asplain format is 1 to 4294967295.
- Range for 4-byte Autonomous system numbers (ASNs) as dot format is 1.0 to 65535.65535.

For the `isis` keyword, an IS-IS instance name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.

For the `ospf` keyword, an OSPF instance name from which routes are to be redistributed. The value takes the form of a string. A decimal number can be entered, but it is stored internally as a string.

Metric metric-value

(Optional) Specifies the metric used for the redistributed route. Range is 1 to 16777214. Use a value consistent with the destination protocol.

Metric-type { 1 | 2 }

(Optional) Specifies the external link type associated with the route advertised into the OSPF routing domain. It can be one of two values:

- **1** — Type 1 external route
- **2** — Type 2 external route

If no `metric-type` is specified, the default is Type 2 external routes.

tag tag-value

(Optional) Specifies the 32-bit dotted-decimal value attached to each external route. This value is not used by the OSPF protocol itself, but is carried in the External LSAs. Range is 0 to 4294967295.

Policy policy-name

(Optional) Specifies the identifier of a configured policy. A policy is used to filter the importation of routes from this source routing protocol to OSPF.

Connected

Distributes routes that are established automatically by virtue of having enabled IP on an interface.

isis

Distributes routes from the IS-IS protocol.

level-1

(Optional) Redistributes Level 1 routes into other IP routing protocols independently.

level-1-2

(Optional) Redistributes both Level 1 and Level 2 routes into other IP routing protocols.

level-2

(Optional) Redistributes Level 2 routes into other IP routing protocols independently.

ospf

Distributes routes from the OSPF protocol.
match { internal | external [ 1 | 2 ] | nssa-external [ 1 | 2 ] }

(Optional) Specifies the criteria by which OSPF routes are redistributed into other routing domains. It can be one or more of the following:

- **internal** — Routes that are internal to a specific autonomous system (intra- and inter-area OSPF routes).
- **external [ 1 | 2 ]** — Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 external routes.
- **nssa-external [ 1 | 2 ]** — Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 not-so-stubby area (NSSA) external routes.

For the **external** and **nssa-external** options, if a type is not specified, then both Type 1 and Type 2 are assumed.

If no match is specified, the default is no filtering.

<table>
<thead>
<tr>
<th>Command Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route redistribution is disabled.</td>
</tr>
</tbody>
</table>

| metric metric-value: | Default is 20 for routes from all protocols except BGP routes, in which the default is 1. |

| metric-type type-value: | Type 2 external route |

All routes from the OSPFv3 routing protocol are redistributed.

| tag tag-value: | If no value is specified, the remote autonomous system number is used for routes from Border Gateway Protocol (BGP); for other protocols, the default is 0. |

<table>
<thead>
<tr>
<th>Command Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router OSPFv3 configuration</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0 This command was introduced.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Usage Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Note</strong> When redistributing routes (into OSPF) using both command keywords for setting or matching of attributes and a route policy, the routes are run through the route policy first, followed by the keyword matching and setting.</td>
</tr>
</tbody>
</table>

Disabling or changing the arguments of any keyword does not affect the state of other keywords.

In general, route redistribution from Level 1 to Level 2 is automatic. You might want to use this command to better control which Level 1 routes can be redistributed into Level 2.

The redistribution of Level 2 routes into Level 1 is called **route leaking**. Route leaking is disabled by default. That is, Level 2 routes are not automatically included in Level 1 link-state protocols. If you want to leak Level 2 routes into Level 1, you must enable that behavior by using this command.

Redistribution from Level 1 into Level 1 and from Level 2 into Level 2 is not allowed.
A router receiving a link-state packet with an internal metric considers the cost of the route from itself to the redistributing router plus the advertised cost to reach the destination. An external metric considers only the advertised metric to reach the destination.

Redistributed routing information should always be filtered by the `distribute-list prefix-list out` command. Use of this command ensures that only those routes intended by the administrator are passed along to the receiving routing protocol.

**OSPFv3 Considerations**

Whenever you use the `redistribute` or the `default-information` command to redistribute routes into an OSPFv3 routing domain, the router automatically becomes an ASBR. However, an ASBR does not, by default, generate a default route into the OSPFv3 routing domain.

When routes are redistributed between OSPFv3 processes, no OSPFv3 metrics are preserved.

When routes are redistributed into OSPF and no metric is specified with the `metric` keyword, OSPF uses 20 as the default metric for routes from all protocols except BGP routes, which get a metric of 1. Furthermore, when the router redistributes from one OSPFv3 process to another OSPFv3 process on the same router, and if no default metric is specified, the metrics in one process are carried to the redistributing process.

**BGP Considerations**

The only connected routes affected by this command are the routes not specified by the `network` (BGP) command.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to cause static routes to be redistributed into an OSPFv3 domain:

```
RP/0/RP0/CP00:router(config)# router ospfv3 109
RP/0/RP0/CP00:router(config-ospfv3)# redistribute isis level-1
```
retransmit-interval (OSPFv3)

To specify the time between link-state advertisement (LSA) retransmissions for adjacencies belonging to the Open Shortest Path First Version 3 (OSPFv3) interface, use the `retransmit-interval` command in an appropriate configuration mode. To return to the default value, use the `no` form of this command.

```
retransmit-interval  seconds
no retransmit-interval
```

**Syntax Description**

- `seconds` Time (in seconds) between retransmissions. It must be greater than the expected round-trip delay between any two routers on the attached network. Range is 1 to 65535 seconds.

**Command Default**

- If this command is not specified in interface configuration mode, then the interface adopts the retransmit interval parameter specified by the area.
- If this command is not specified in area configuration mode, then the interface adopts the retransmit interval parameter specified by the process.
- If this command is not specified at any level, then the default retransmit interval is 5 seconds.

**Command Modes**

- Interface configuration
- Area configuration
- Router OSPFv3 configuration
- Virtual-link configuration

**Command History**

```
Release 6.0  This command was introduced.
```

**Usage Guidelines**

- When a router sends an LSA to its neighbor, it keeps the LSA until it receives the acknowledgment message. If the router receives no acknowledgment, it resends the LSA.
- The setting of this parameter should be conservative, or needless retransmission results. The value should be larger for serial lines and virtual links.

**Task ID**

```
Task ID      Operations
------------
ospf         read, write
```

**Examples**

The following example shows how to set the retransmit interval value to 8 seconds while in interface configuration mode:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/2/0/0
```
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# retransmit-interval 8
router-id (OSPFv3)

To configure a router ID for the Open Shortest Path First Version 3 (OSPFv3) routing process, use the `router-id` command in an appropriate configuration mode. To cause the software to use the default method of determining the router ID, use the `no` form of this command after clearing or restarting the OSPF process.

```
router-id router-id
no router-id router-id
```

**Syntax Description**

`router-id` 32-bit router ID value specified in four-part, dotted-decimal notation.

**Command Default**

If this command is not configured, the router ID is the highest IP address for an interface on the router, with any loopback interface taking precedence.

**Command Modes**

Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

We recommend that you use the `router-id` command to explicitly specify a unique 32-bit numeric value for the router ID. This configuration ensures that OSPFv3 can function regardless of the interface address configuration. Clear the OSPF process using the `clear ospf process` command or restart the OSPF process for the `no router-id` command to take effect.

In router OSPFv3 configuration mode, OSPF attempts to obtain a router ID in the following ways (in order of preference):

1. By default, when the OSPF process initializes, it checks if there is a router-id in the checkpointing database.
2. The 32-bit numeric value specified by the OSPF `router-id` command in router configuration mode. (This value can be any 32-bit value. It is not restricted to the IPv4 addresses assigned to interfaces on this router, and need not be a routable IPv4 address.)
3. A global router ID provided by the system (possibly, the first loopback address found at the boot time.)

If the OSPFv3 process cannot obtain a router ID from any of these sources, the router issues the following error message:

```
%OSPFv3-4-NORTRID : OSPFv3 process 1 cannot run - configure a router ID for this process
```

At this point, OSPFv3 is effectively passive on all its interfaces. To run OSPFv3, make a router ID available by one of the methods described.
The following example shows how to assign the IP address of 10.0.0.10 to the OSPFv3 process 109:

```bash
RP/0/RP0/CPU0:router(config)# router ospfv3 109
RP/0/RP0/CPU0:router(config-ospfv3)# router-id 10.0.0.10
```
To configure an Open Shortest Path First Version 3 (OSPFv3) routing process, use the `router ospfv3` command in XR Config mode. To terminate an OSPFv3 routing process, use the `no` form of this command.

```
router ospfv3 process-name
no router ospfv3 process-name
```

**Syntax Description**
- `process-name` Name that uniquely identifies an OSPFv3 routing process. The process name is any alphanumeric string no longer than 40 characters.

**Command Default**
No OSPFv3 routing process is defined.

**Command Modes**
XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
You can specify multiple OSPFv3 routing processes in each router. Up to 10 processes can be configured. The recommendation is not to exceed 4 OSPFv3 processes.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to instantiate an OSPFv3 routing process with a process name of 1:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
```
sham-link (OSPFv3)

To configure an Open Shortest Path First version 3 (OSPFv3) sham link between two provider edge routers on a non-default VRF, use the sham-link command in OSPFv3 router area sub-configuration mode. To terminate an OSPFv3 sham link, use the no form of this command.

```
sham-link source-address destination-address
no sham-link
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Source-Address</th>
<th>IP address of the local (source) sham-link endpoint specified in four-part, dotted-decimal notation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Destination-Address</td>
<td>IP address of the remote (destination) sham-link endpoint specified in four-part, dotted-decimal notation.</td>
</tr>
</tbody>
</table>

**Command Default**

No sham link is configured.

**Command Modes**

OSPFv3 router area sub-configuration.

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the sham-link command to configure a point-to-point connection between two provider edge (PE) routers creating an interconnect between two VPN sites (VPN backbone). Sham links are configured on PE provider edge (PE) routers in a Multiprotocol Label Switching (MPLS) VPN backbone.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

This example shows how to configure an OSPFv3 sham link:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router ospfv3 ospfv3
RP/0/RP0/CPU0:router(config_ospfv3)# vrf 1
RP/0/RP0/CPU0:router(config_ospfv3_vrf)# area 1
RP/0/RP0/CPU0:router(config_ospfv3_vrf_ar)# sham-link 100::1 200::1
RP/0/RP0/CPU0:router(config_ospfv3_vrf_ar_sl)# cost 23
```
show ospfv3

To display general information about Open Shortest Path First Version 3 (OSPFv3) routing processes, use the `show ospfv3` command in XR EXEC mode.

**show ospfv3 [process-name] sham-links**

**Syntax Description**

- **process-name** (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.

- **sham-links** Displays OSPFv3 Sham-link information.

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

- **Release**
  - 6.0  This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Examples**

The following is a sample output from the `show ospfv3` command:

```
RP/0/RP0/CPU0:router# show ospfv3 1
Routing Process "ospfv3 test" with ID 3.3.3.3
It is an autonomous system boundary router
Redistributing External Routes from, static
  Maximum number of redistributed prefixes 10240
  Threshold for warning message 75%
  Initial SPF schedule delay 5000 msecs
  Minimum hold time between two consecutive SPFcs 10000 msecs
  Maximum wait time between two consecutive SPFcs 10000 msecs
  Initial LSA throttle delay 0 msecs
  Minimum hold time for LSA throttle 5000 msecs
  Maximum wait time for LSA throttle 5000 msecs
  Minimum LSA arrival 1000 msecs
  LSA group pacing timer 240 secs
  Interface flood pacing timer 33 msecs
  Retransmission pacing timer 66 msecs
  Maximum number of configured interfaces 255
  Number of external LSA 1. Checksum Sum 0x004468
  Number of areas in this router is 1. 1 normal 0 stub 0 nssa
```
Area BACKBONE(0) (Inactive)
Number of interfaces in this area is 1
SPF algorithm executed 1 times
Number of LSA 3. Checksum Sum 0x018109
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0

The following is a sample output from the **show ospfv3** command when OSPFv3 graceful shutdown has been initiated but not yet completed:

```
RP/0/RP0/CPU0:router# show ospfv3 1
Routing Process "ospfv3 test" with ID 3.3.3.3 transitioning to OSPFV3_ADMIN_DOWN state
Routing Process "ospfv3 test" with ID 3.3.3.3
It is an autonomous system boundary router
Redistributing External Routes from,
static
Maximum number of redistributed prefixes 10240
Threshold for warning message 75%
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPFs 10000 msecs
Maximum wait time between two consecutive SPFs 10000 msecs
Initial LSA throttle delay 0 msecs
Minimum hold time for LSA throttle 5000 msecs
Maximum wait time for LSA throttle 5000 msecs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Maximum number of configured interfaces 255
Number of external LSA 1. Checksum Sum 0x004468
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Area BACKBONE(0) (Inactive)
Number of interfaces in this area is 1
SPF algorithm executed 1 times
Number of LSA 3. Checksum Sum 0x018109
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0
```

The following is a sample output from the **show ospfv3** command when OSPFv3 graceful shutdown is completed:

```
RP/0/RP0/CPU0:router# show ospfv3 1
Routing Process "ospfv3 test" with ID 3.3.3.3 in OSPFV3_ADMIN_DOWN state
%ROUTING-OSPFV3-6-GRACEFUL_SHUTDOWN : Shutdown Complete
Routing Process "ospfv3 test" with ID 3.3.3.3
It is an autonomous system boundary router
Redistributing External Routes from,
static
Maximum number of redistributed prefixes 10240
Threshold for warning message 75%
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPFs 10000 msecs
Maximum wait time between two consecutive SPFs 10000 msecs
```

Routing Command Reference for Cisco NCS 5000 Series Routers
Initial LSA throttle delay 0 msecs
Minimum hold time for LSA throttle 5000 msecs
Maximum wait time for LSA throttle 5000 msecs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Maximum number of configured interfaces 255
Number of external LSA 1. Checksum Sum 0x004468
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Area BACKBONE(0) (Inactive)
Number of interfaces in this area is 1
SPF algorithm executed 1 times
Number of LSA 3. Checksum Sum 0x018109
Number of DCbitless LSA 0
Number of indication LSA 0
Number of DoNotAge LSA 0
Flood list length 0

This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing Process “ospfv3 test” with ID</td>
<td>OSPFv3 process name.</td>
</tr>
<tr>
<td>It is</td>
<td>Types are internal, area border, or autonomous system boundary.</td>
</tr>
<tr>
<td>Redistributing External Routes from</td>
<td>Lists of redistributed routes, by protocol.</td>
</tr>
<tr>
<td>Maximum number of redistributed prefixes</td>
<td>Number of redistributed prefixes</td>
</tr>
<tr>
<td>Threshold for warning message</td>
<td>Warning message threshold.</td>
</tr>
<tr>
<td>Initial SPF schedule delay</td>
<td>Delay time of SPF calculations.</td>
</tr>
<tr>
<td>Minimum hold time between two consecutive SPFs</td>
<td>Minimum hold time between consecutive SPFs.</td>
</tr>
<tr>
<td>Maximum wait time between two consecutive SPFs</td>
<td>Maximum wait time between consecutive SPFs.</td>
</tr>
<tr>
<td>Initial LSA throttle delay</td>
<td>Delay time of LSA throttle.</td>
</tr>
<tr>
<td>Maximum hold time for LSA throttle</td>
<td>After initial throttle delay, the LSA generation is backed off by hold interval.</td>
</tr>
<tr>
<td>Maximum wait time for LSA throttle</td>
<td>Maximum throttle delay for LSA generation.</td>
</tr>
<tr>
<td>Minimum LSA arrival</td>
<td>Minimum LSA arrival.</td>
</tr>
<tr>
<td>LSA group pacing timer</td>
<td>Configured LSA group pacing timer (in seconds).</td>
</tr>
<tr>
<td>Interface flood pacing timer</td>
<td>Flooding pacing interval.</td>
</tr>
<tr>
<td>Retransmission pacing timer</td>
<td>Retransmission pacing interval.</td>
</tr>
<tr>
<td>Maximum number of configured interfaces</td>
<td>Maximum number of configured interfaces.</td>
</tr>
</tbody>
</table>
This is sample output from the show ospfv3 vrf command that displays domain -id configuration:

```
RF/0/RP0/CPU0:router#show ospfv3 0 vrf V1
Mon May 10 14:52:31.332 CEST

Routing Process "ospfv3 0" with ID 100.0.0.2 VRF V1
It is an area border and autonomous system boundary router
Redistributing External Routes from,
bgp 1
Maximum number of redistributed prefixes 10240
Threshold for warning message 75%
Primary Domain ID:
  0x0005:0xc0fe00112233
Secondary Domain ID:
  0x0105:0xbeef00000001
  0x0205:0xbeef00000002
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPFSS 10000 msecs
Maximum wait time between two consecutive SPFSS 10000 msecs
Initial LSA throttle delay 0 msecs
Minimum hold time for LSA throttle 5000 msecs
Maximum wait time for LSA throttle 5000 msecs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Maximum number of configured interfaces 255
Maximum number of configured paths 16
Number of external LSA 2. Checksum Sum 0x0f15bb3
Number of areas in this router is 1. 1 normal 0 stub 0 nssa
Auto cost is enabled. Reference bandwidth 100
Area BACKBONE(0)
  Number of interfaces in this area is 1
  SPF algorithm executed 2 times
  Number of LSA 4. Checksum Sum 0x02629d
  Number of DCbitless LSA 0
  Number of indication LSA 0
  Number of DoNotAge LSA 0
  Flood list length 0
```
This is sample output from the show ospfv3 vrf command that displays vrf-lite configuration:

```
RP/0/RP0/CPU0:router# show ospfv3 0 vrf V2
Mon May 10 18:01:38.654 CEST
Routing Process "ospfv3 0" with ID 2.2.2.2 VRF V2
VRF lite capability is enabled
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPF s 10000 msecs
Maximum wait time between two consecutive SPF s 10000 msecs
Initial LSA throttle delay 0 msecs
Minimum hold time for LSA throttle 5000 msecs
Maximum wait time for LSA throttle 5000 msecs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Maximum number of configured interfaces 255
Maximum number of configured paths 16
Number of external LSA 0. Checksum Sum 00000000
Number of areas in this router is 0. 0 normal 0 stub 0 nssa
Auto cost is enabled. Reference bandwidth 100
```

This is sample output from the `show ospfv3` command to verify that (Non-stop routing (NSR) is enabled:

```
RP/0/RP0/CPU0:router# show ospfv3
Routing Process "ospfv3 100" with ID 3.3.3.3
NSR (Non-stop routing) is Enabled
It is an area border and autonomous system boundary router
Redistributing External Routes from,
    bgp 100
    Maximum number of redistributed prefixes 10240
    Threshold for warning message 75%
Initial SPF schedule delay 5000 msecs
Minimum hold time between two consecutive SPF s 10000 msecs
Maximum wait time between two consecutive SPF s 10000 msecs
Initial LSA throttle delay 0 msecs
Minimum hold time for LSA throttle 5000 msecs
Maximum wait time for LSA throttle 5000 msecs
Minimum LSA arrival 1000 msecs
LSA group pacing timer 240 secs
Interface flood pacing timer 33 msecs
Retransmission pacing timer 66 msecs
Maximum number of configured interfaces 512
Maximum number of configured paths 16
Number of external LSA 0. Checksum Sum 00000000
Number of areas in this router is 15. 15 normal 0 stub 0 nssa
Auto cost is enabled. Reference bandwidth 100
```

The following is a sample output from the `show ospfv3` command with `sham-links` keyword:

```
RP/0/RP0/CPU0:router# show ospfv3 vrf vrf1 sham-links
Sham Links for OSPFV3 100, VRF vrf1
Sham Link OSPF SL1 to address 300::1 is up
Area 2, source address 100::1
IfIndex = 2
```
Run as demand circuit
DoNotAge LSA allowed., Cost of using 1
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 00:00:00
Adjacency State FULL (Hello suppressed)
Number of DBD retrans during last exchange 0
Index 2/2, retransmission queue length 0, number of retransmission 0
First 0(0)/0(0) Next 0(0)/0(0)
Last retransmission scan length is 0, maximum is 0
Last retransmission scan time is 0 msec, maximum is 0 msec
Sham Link OSPF_SL0 to address 200::1 is up
Area 2, source address 100::1
IfIndex = 2
Run as demand circuit
DoNotAge LSA allowed., Cost of using 1
Transmit Delay is 1 sec, State POINT_TO_POINT,
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 00:00:01
Adjacency State FULL (Hello suppressed)
Number of DBD retrans during last exchange 0
Index 3/3, retransmission queue length 0, number of retransmission 0
First 0(0)/0(0) Next 0(0)/0(0)
Last retransmission scan length is 0, maximum is 0
Last retransmission scan time is 0 msec, maximum is 0 msec
show ospfv3 border-routers

To display the internal Open Shortest Path First Version 3 (OSPFv3) routing table entries to an area border router (ABR) and autonomous system boundary router (ASBR), use the `show ospfv3 border-routers` command in XR EXEC mode.

```
show ospfv3 [process-name] border-routers [router-id]
```

**Syntax Description**

- `process-name` (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.
- `router-id` (Optional) 32-bit router ID value specified in four-part, dotted-decimal notation.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show ospfv3 border-routers` command:

```
RP/0/RP0/CPU0:router# show ospfv3 border-routers

OSPFv3 1 Internal Routing Table
Codes: i - Intra-area route, I - Inter-area route
i 10.0.0.207 [1] via fe80::3034:30ff:fe33:3742, tenGigE 0/3/0/0, ABR/ASBR, Area 1, SPF 3
i 10.0.0.207 [10] via fe80::204:c0ff:fe22:73fe, Ethernet0/0/0/0, ABR/ASBR, Area 0, SPF 7
```

This table describes the significant fields shown in the display.

**Table 62: show ospf border-routers Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>Type of this route; i indicates an intra-area route, I an inter-area route.</td>
</tr>
<tr>
<td>10.0.0.207</td>
<td>Router ID of destination.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>[1]</td>
<td>Cost of using this route.</td>
</tr>
<tr>
<td>fe80::3034:30ff:fe33:3742</td>
<td>Next-hop toward the destination.</td>
</tr>
<tr>
<td>tenGigE 0/3/0/0</td>
<td>Packets destined for fe80::3034:30ff:fe33:3742 are sent over tenGigE interface 3/0/0/0.</td>
</tr>
<tr>
<td>ABR/ASBR</td>
<td>Router type of the destination; it is either an area border router (ABR) or autonomous system boundary router (ASBR) or both.</td>
</tr>
<tr>
<td>Area 1</td>
<td>Area ID of the area from which this route is learned.</td>
</tr>
<tr>
<td>SPF 3</td>
<td>Internal number of the shortest path first (SPF) calculation that installs this route.</td>
</tr>
</tbody>
</table>
show ospfv3 database

To display lists of information related to the Open Shortest Path First Version 3 (OSPFv3) database for a specific router, use the `show ospfv3 database` command in XR EXEC mode.

```
show ospfv3 [process-name [area-id]] database
show ospfv3 [process-name [area-id]] database [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [database-summary]
show ospfv3 [process-name [area-id]] database [external] [link-state-id]
show ospfv3 [process-name [area-id]] database [external] [link-state-id] [internal] [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [external] [link-state-id] [internal] [self-originate]
show ospfv3 [process-name [area-id]] database [inter-area prefix] [link-state-id]
show ospfv3 [process-name [area-id]] vrf vrf-name database [link-state-id] [internal] [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [inter-area prefix] [link-state-id] [internal]
show ospfv3 [process-name [area-id]] database [inter-area router] [link-state-id] [internal]
show ospfv3 [process-name [area-id]] database [link] [link-state-id] [internal]
show ospfv3 [process-name [area-id]] database [link] [link-state-id] [internal] [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [link] [link-state-id] [internal] [self-originate]
show ospfv3 [process-name [area-id]] database [network] [link-state-id]
show ospfv3 [process-name [area-id]] database [network] [link-state-id] [internal] [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [network] [link-state-id] [internal] [self-originate]
show ospfv3 [process-name [area-id]] database [nssa-external] [link-state-id]
show ospfv3 [process-name [area-id]] database [nssa-external] [link-state-id] [internal] [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [nssa-external] [link-state-id] [internal] [self-originate]
show ospfv3 [process-name [area-id]] database [prefix] [ref-lsa] [link-state-id]
show ospfv3 [process-name [area-id]] database [prefix] [ref-lsa] [link-state-id] [internal] [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [prefix] [ref-lsa] [link-state-id] [internal] [self-originate]
show ospfv3 [process-name [area-id]] database [prefix] [link-state-id]
show ospfv3 [process-name [area-id]] database [prefix] [link-state-id] [internal] [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [prefix] [link-state-id] [internal] [self-originate]
show ospfv3 [process-name [area-id]] database [router] [link-state-id]
show ospfv3 [process-name [area-id]] database [router] [link-state-id] [internal]
show ospfv3 [process-name [area-id]] database [router] [adv-router [router-id]]
show ospfv3 [process-name [area-id]] database [router] [link-state-id] [internal] [self-originate]
show ospfv3 [process-name [area-id]] database [self-originate]
```
### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>process-name</code></td>
<td>(Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the <code>router ospfv3</code> command. If this argument is included, only information for the specified routing process is displayed.</td>
</tr>
<tr>
<td><code>area-id</code></td>
<td>(Optional) Area number used to define the particular area.</td>
</tr>
<tr>
<td><code>adv-router [router-id]</code></td>
<td>(Optional) Displays all link-state advertisements (LSAs) of the specified router.</td>
</tr>
<tr>
<td><code>asbr-summary</code></td>
<td>(Optional) Displays information only about the Autonomous System Boundary Router (ASBR) summary LSAs.</td>
</tr>
<tr>
<td><code>database-summary</code></td>
<td>(Optional) Displays how many of each type of LSA are in the database for each area and the total.</td>
</tr>
<tr>
<td><code>external</code></td>
<td>(Optional) Displays information only about external LSAs.</td>
</tr>
<tr>
<td><code>grace</code></td>
<td>(Optional) Displays information about the state for the graceful restart link.</td>
</tr>
<tr>
<td><code>internal</code></td>
<td>(Optional) Displays information only about internal LSAs.</td>
</tr>
<tr>
<td><code>self-originate</code></td>
<td>(Optional) Displays only self-originated LSAs (from the local router).</td>
</tr>
<tr>
<td><code>link-state-id</code></td>
<td>(Optional) LSA ID that uniquely identifies the LSA. For network LSAs and link LSAs, this ID is the interface ID of the link of the router originating the LSA.</td>
</tr>
<tr>
<td><code>inter-area prefix</code></td>
<td>(Optional) Displays information only about the interarea prefix LSAs.</td>
</tr>
<tr>
<td><code>inter-area router</code></td>
<td>(Optional) Displays information only about the interarea router LSAs.</td>
</tr>
<tr>
<td><code>link</code></td>
<td>(Optional) Displays information only about the link LSAs.</td>
</tr>
<tr>
<td><code>network</code></td>
<td>(Optional) Displays information only about the network LSAs.</td>
</tr>
<tr>
<td><code>nssa-external</code></td>
<td>(Optional) Displays information only about the not-so-stubby area (NSSA) external LSAs.</td>
</tr>
<tr>
<td><code>prefix</code></td>
<td>(Optional) Displays information only about the prefix LSAs.</td>
</tr>
<tr>
<td><code>ref-lsa</code></td>
<td>(Optional) Displays referenced LSA information.</td>
</tr>
<tr>
<td><code>router</code></td>
<td>(Optional) Displays information only about the router LSAs.</td>
</tr>
<tr>
<td><code>unknown</code></td>
<td>(Optional) Displays information only about unknown LSAs.</td>
</tr>
<tr>
<td><code>area</code></td>
<td>(Optional) Displays information only about the area LSAs.</td>
</tr>
<tr>
<td><code>as</code></td>
<td>(Optional) Displays information only about the autonomous system LSAs.</td>
</tr>
</tbody>
</table>

### Command Default

No default behavior or values

### Command Modes

XR EXEC mode
**Command History**

**Release**  
Modification

- Release 6.0  This command was introduced.

**Usage Guidelines**

The various forms of this command deliver information about different OSPFv3 link-state advertisements.

**Examples**

The following is sample output from the `show ospfv3 database` command with no arguments or keywords:

```
RP/0/RP0/CPU0:router# show ospfv3 database

OSPFv3 Router with ID (10.0.0.207) (Process ID 1)

Router Link States (Area 0)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Fragment ID</th>
<th>Link count</th>
<th>Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.0.1</td>
<td>163</td>
<td>0x80000039</td>
<td>0</td>
<td>2</td>
<td>None</td>
</tr>
<tr>
<td>10.0.0.206</td>
<td>145</td>
<td>0x80000005</td>
<td>0</td>
<td>1</td>
<td>EB</td>
</tr>
<tr>
<td>10.0.0.207</td>
<td>151</td>
<td>0x80000004</td>
<td>0</td>
<td>1</td>
<td>EB</td>
</tr>
<tr>
<td>192.168.0.0</td>
<td>163</td>
<td>0x80000039</td>
<td>0</td>
<td>1</td>
<td>None</td>
</tr>
</tbody>
</table>

Net Link States (Area 0)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Link ID</th>
<th>Rtr count</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.207</td>
<td>152</td>
<td>0x80000002</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>192.168.0.0</td>
<td>163</td>
<td>0x80000039</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

Inter Area Prefix Link States (Area 0)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Prefix</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.206</td>
<td>195</td>
<td>0x80000001</td>
<td>3002::/56</td>
</tr>
<tr>
<td>10.0.0.207</td>
<td>197</td>
<td>0x80000001</td>
<td>3002::/56</td>
</tr>
<tr>
<td>10.0.0.206</td>
<td>195</td>
<td>0x80000001</td>
<td>3002::206/128</td>
</tr>
<tr>
<td>10.0.0.207</td>
<td>182</td>
<td>0x80000001</td>
<td>3002::206/128</td>
</tr>
</tbody>
</table>

Inter Area Router Link States (Area 0)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Link ID</th>
<th>Dest RtrID</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.207</td>
<td>182</td>
<td>0x80000001</td>
<td>167772366</td>
<td>10.0.0.206</td>
</tr>
<tr>
<td>10.0.0.206</td>
<td>182</td>
<td>0x80000001</td>
<td>167772367</td>
<td>10.0.0.207</td>
</tr>
</tbody>
</table>

Link (Type-8) Link States (Area 0)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Link ID</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.0.1</td>
<td>163</td>
<td>0x80000039</td>
<td>1</td>
<td>Et0/0/0/0</td>
</tr>
<tr>
<td>10.0.0.207</td>
<td>202</td>
<td>0x80000001</td>
<td>1</td>
<td>Et0/0/0/0</td>
</tr>
<tr>
<td>10.0.0.206</td>
<td>200</td>
<td>0x80000001</td>
<td>2</td>
<td>Et0/0/0/0</td>
</tr>
</tbody>
</table>

Intra Area Prefix Link States (Area 0)

<table>
<thead>
<tr>
<th>ADV Router</th>
<th>Age</th>
<th>Seq#</th>
<th>Link ID</th>
<th>Ref-lstype</th>
<th>Ref-LSID</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.0.0</td>
<td>163</td>
<td>0x80000039</td>
<td>0</td>
<td>0x2002</td>
<td>1</td>
</tr>
<tr>
<td>192.168.0.0</td>
<td>163</td>
<td>0x80000039</td>
<td>1</td>
<td>0x2001</td>
<td>0</td>
</tr>
</tbody>
</table>
```
This table describes the significant fields shown in the display.

**Table 63: show ospfv3 database Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADV Router</td>
<td>ID of advertising router.</td>
</tr>
<tr>
<td>Age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Seq#</td>
<td>Link-state sequence number (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Fragment ID</td>
<td>Router LSA fragment ID.</td>
</tr>
<tr>
<td>Link count</td>
<td>Number of links described.</td>
</tr>
<tr>
<td>Bits</td>
<td>B indicates that the router is an area border router. E indicates that the router is an autonomous system boundary router. V indicates that the router is a virtual link endpoint. W indicates that the router is a wildcard multicast receiver.</td>
</tr>
<tr>
<td>Link ID</td>
<td>Unique LSA ID.</td>
</tr>
<tr>
<td>Rtr count</td>
<td>Number of routers attached to the link.</td>
</tr>
<tr>
<td>Prefix</td>
<td>Prefix of the route being described.</td>
</tr>
<tr>
<td>Dest RtrID</td>
<td>Router ID of the router being described.</td>
</tr>
<tr>
<td>Interface</td>
<td>Link described by the LSA.</td>
</tr>
<tr>
<td>Ref-lsType</td>
<td>LSA type of the LSA being referenced.</td>
</tr>
<tr>
<td>Ref-LSID</td>
<td>LSA ID of the LSA being referenced.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospfv3 database` command with the `external` keyword:

```
RP/0/RP0/CPU0:router# show ospfv3 database external

OSPFv3 Router with ID (10.0.0.206) (Process ID 1)

Type-5 AS External Link States

  LS age: 189
  LS Type: AS External Link
  Link State ID: 0
  Advertising Router: 10.0.0.206
  LS Seq Number: 80000002
  Checksum: 0xa303
  Length: 36
  Prefix Address: 2222::
  Prefix Length: 56, Options: None
  Metric Type: 2 (Larger than any link state path)
```
This table describes the significant fields shown in the display.

**Table 64: show ospfv3 database external Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPFv3 Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPFv3 process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of Advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence number (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>LS checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Prefix Address</td>
<td>IPv6 address prefix of the route being described.</td>
</tr>
<tr>
<td>Prefix Length</td>
<td>Length of the IPv6 address prefix.</td>
</tr>
<tr>
<td>Metric Type</td>
<td>External type.</td>
</tr>
<tr>
<td>Metric</td>
<td>Link-state metric.</td>
</tr>
<tr>
<td>External Route Tag</td>
<td>External route tag, a 32-bit field attached to each external route. This tag is not used by the OSPFv3 protocol itself.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospfv3 database` command with the `inter-area prefix` keyword:

```
RP/0/RP0/CPU0:router# show ospfv3 database inter-area prefix

OSPFv3 Router with ID (10.0.0.206) (Process ID 1)

Inter Area Prefix Link States (Area 0)

LS age: 715
LS Type: Inter Area Prefix Links
Link State ID: 0
Advertising Router: 10.0.0.206
LS Seq Number: 80000002
Checksum: 0x3cb5
Length: 36
Metric: 1
Prefix Address: 3002::
Prefix Length: 56, Options: None
```
This table describes the significant fields shown in the display.

*Table 65: show ospfv3 database inter-area prefix Field Descriptions*

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPFv3 Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPFv3 process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Metric</td>
<td>Link-state metric.</td>
</tr>
<tr>
<td>Prefix Address</td>
<td>IPv6 prefix of the route being described.</td>
</tr>
<tr>
<td>Prefix Length</td>
<td>IPv6 prefix length of the route being described.</td>
</tr>
<tr>
<td>Options</td>
<td>LA indicates that the prefix is a local address. MC indicates the prefix is multicast capable. NU indicates that the prefix is not unicast capable. P indicates that the prefix should be propagated at a not-so-stubby area (NSSA) area border.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospfv3 database` command with the `inter-area router` keyword:

```
RP/0/RP0/CPU0:router# show ospfv3 database inter-area router

OSPFv3 Router with ID (10.0.0.206) (Process ID 1)

    Inter Area Router Link States (Area 0)

        LS age: 1522
        Options: (V6-Bit E-Bit R-bit DC-Bit)
        LS Type: Inter Area Router Links
        Link State ID: 167772366
        Advertising Router: 10.0.0.207
        LS Seq Number: 80000002
        Checksum: 0xcaae
        Length: 32
        Metric: 1
        Destination Router ID: 10.0.0.206
```
This table describes the significant fields shown in the display.

**Table 66: show ospfv3 database inter-area routerField Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPFv3 Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPFv3 process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only): DC—Supports demand circuits. E—Capable of processing external LSAs. MC—Forwards IP multicast. N—Supports Type 7 LSAs. R—Router is active. V6—Include in IPv6 routing calculations.</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA.)</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSAs.</td>
</tr>
<tr>
<td>Metric</td>
<td>Link-state metric.</td>
</tr>
<tr>
<td>Destination Router ID</td>
<td>Router ID of the router being described.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospfv3 database` command with the `link` keyword:

```
RP/0/RP0/CPU0:router# show ospfv3 database link
OSPFv3 Router with ID (10.0.0.206) (Process ID 1)
Link (Type-8) Link States (Area 0)

  LS age: 620
  Options: (V6-Bit E-Bit R-bit DC-Bit)
  LS Type: Link-LSA (Interface: Ethernet0/0/0/0)
  Link State ID: 1 (Interface ID)
  Advertising Router: 10.0.0.207
  LS Seq Number: 80000003
  Checksum: 0x7235
  Length: 56
  Router Priority: 1
  Link Local Address: fe80::204:c0ff:fe22:73fe
  Number of Prefixes: 1
  Prefix Address: 7002::
  Prefix Length: 56, Options: None
```
This table describes the significant fields shown in the display.

**Table 67: show ospfv3 database link Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPFv3 Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID</td>
<td>OSPFv3 process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only):</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID (Interface ID).</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSAs.</td>
</tr>
<tr>
<td>Router Priority</td>
<td>Interface priority of originating router.</td>
</tr>
<tr>
<td>Link Local Address</td>
<td>Link local address of the interface.</td>
</tr>
<tr>
<td>Number of Prefixes</td>
<td>Number of prefixes associated with the link.</td>
</tr>
<tr>
<td>Prefix Address and Length</td>
<td>List of prefixes associated with the link.</td>
</tr>
<tr>
<td>Options</td>
<td>LA indicates that the prefix is a local address. MC indicates that the prefix is multicast capable. NU indicates that the prefix is not unicast capable. P indicates that the prefix should be propagated at an NSSA area border.</td>
</tr>
</tbody>
</table>

The following is sample output from the **show ospfv3 database** command with the **network** keyword:

```
RP/0/RP0/CPU0:router# show ospfv3 database network

OSPFv3 Router with ID (10.0.0.206) (Process ID 1)

Net Link States (Area 0)

  LS age: 1915
  Options: (V6-Bit E-Bit R-bit DC-Bit)
  LS Type: Network Links
  Link State ID: 1 (Interface ID of Designated Router)
  Advertising Router: 10.0.0.207
  LS Seq Number: 80000004
```
This table describes the significant fields shown in the display.

Table 68: show ospfv3 database network Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPFv3 Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID 1</td>
<td>OSPFv3 process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only):</td>
</tr>
<tr>
<td></td>
<td>DC—Supports demand circuits.</td>
</tr>
<tr>
<td></td>
<td>E—Capable of processing external LSAs.</td>
</tr>
<tr>
<td></td>
<td>MC—Forwards IP multicast.</td>
</tr>
<tr>
<td></td>
<td>N—Supports Type 7 LSAs.</td>
</tr>
<tr>
<td></td>
<td>R—Router is active.</td>
</tr>
<tr>
<td></td>
<td>V6—Include in IPv6 routing calculations.</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID of the designated router.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Attached Router</td>
<td>List of routers attached to the network, by router ID.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospfv3 database` command with the `prefix` keyword:

```
RP/0/RP0/CPU0:router# show ospfv3 database prefix
                OSPFv3 Router with ID (10.0.0.206) (Process ID 1)
                  Intra Area Prefix Link States (Area 1)

Routing Bit Set on this LSA
LS age: 356
LS Type: Intra-Area-Prefix-LSA
Link State ID: 0
Advertising Router: 10.0.0.206
LS Seq Number: 8000001e
Checksum: 0xcdaa
Length: 44
Referenced LSA Type: 2001
Referenced Link State ID: 0
Referenced Advertising Router: 10.0.0.206
```
Number of Prefixes: 1  
Prefix Address: 8006::  
Prefix Length: 56, Options: None, Metric: 1

This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPFv3 Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID 1</td>
<td>OSPFv3 process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID of the designated router.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Referenced LSA Type</td>
<td>Router LSA or network LSA of the prefixes referenced.</td>
</tr>
<tr>
<td>Referenced Link State ID</td>
<td>Link-state ID of the router or network LSA.</td>
</tr>
<tr>
<td>Referenced Advertising Router</td>
<td>Advertising router of the referenced LSA.</td>
</tr>
<tr>
<td>Number of Prefixes</td>
<td>Number of prefixes listed in the LSA.</td>
</tr>
<tr>
<td>Prefix Address</td>
<td>Prefix associated with the router or network.</td>
</tr>
<tr>
<td>Prefix Length</td>
<td>Length of the prefix.</td>
</tr>
<tr>
<td>Options</td>
<td>LA indicates that the prefix is a local address. MC indicates that the prefix is multicast capable. NU indicates that the prefix is not unicast capable. P indicates the prefix should be propagated at an NSSA area border.</td>
</tr>
<tr>
<td>Metric</td>
<td>Cost of the prefix.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show ospfv3 database` command with the `router` keyword:

```
RP/0/RP0/CPU0:router# show ospfv3 database router

OSPFv3 Router with ID (10.0.0.206) (Process ID 1)
Router Link States (Area 0)
```
This table describes the significant fields shown in the display.

**Table 70: show ospfv3 database router Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OSPFv3 Router with ID</td>
<td>Router ID number.</td>
</tr>
<tr>
<td>Process ID 1</td>
<td>OSPFv3 process name.</td>
</tr>
<tr>
<td>LS age</td>
<td>Link-state age.</td>
</tr>
<tr>
<td>Options</td>
<td>Type of service options (Type 0 only): &lt;br&gt;DC—Supports demand circuits. E—Capable of processing external LSAs. MC—Forwards IP multicast. N—Supports Type 7 LSAs. R—Router is active. V6—Include in IPv6 routing calculations.</td>
</tr>
<tr>
<td>LS Type</td>
<td>Link-state type.</td>
</tr>
<tr>
<td>Link State ID</td>
<td>Link-state ID of the designated router.</td>
</tr>
<tr>
<td>Advertising Router</td>
<td>ID of the advertising router.</td>
</tr>
<tr>
<td>LS Seq Number</td>
<td>Link-state sequence (detects old or duplicate LSAs).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Link-state checksum (Fletcher checksum of the complete contents of the LSA).</td>
</tr>
<tr>
<td>Length</td>
<td>Length (in bytes) of the LSA.</td>
</tr>
<tr>
<td>Link connected to</td>
<td>The type of network to which this interface is connected. Values are: &lt;br&gt;• Another Router (point-to-point). &lt;br&gt;• A Transit Network. &lt;br&gt;• A Virtual Link.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>Link Metric</td>
<td>OSPF cost of using this link.</td>
</tr>
<tr>
<td>Local Interface ID</td>
<td>Number that uniquely identifies an interface on a router.</td>
</tr>
</tbody>
</table>
show ospfv3 flood-list

To display a list of Open Shortest Path First Version 3 (OSPFv3) link-state advertisements (LSAs) waiting to be flooded over an interface, use the `show ospfv3 flood-list` command in XR EXEC mode.

```
show ospfv3 [process-name] [area-id] flood-list [type interface-path-id]
```

**Syntax Description**

- `process-name` (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.

- `area-id` (Optional) Area number used to define the particular area.

- `type` Interface type. For more information, use the question mark (?) online help function.

- `interface-path-id` Physical interface or virtual interface.

**Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release 6.0 | This command was introduced. |

**Usage Guidelines**

Use the `show ospfv3 flood-list` command to display OSPFv3 packet pacing.

**Examples**

The following sample output from the `show ospfv3 flood-list` command shows three entries for the OSPFv3 1 process running over tenGigE interface 0/3/0/0:

```
RP/0/RP0/CPU0:router# show ospfv3 flood-list tenGigE 0/3/0/0

Flood Lists for OSPFV3 1

  Interface tenGigE 0/3/0/0, Queue length 3
    Link state retransmission due in 24 msec

  Displaying 3 entries from flood list:
```

Routing Command Reference for Cisco NCS 5000 Series Routers
This table describes the significant fields shown in the display.

Table 71: show ospfv3 flood-list Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
<td>Interface for which information is displayed.</td>
</tr>
<tr>
<td>Queue length</td>
<td>Number of LSAs waiting to be flooded.</td>
</tr>
<tr>
<td>Link state retransmission due in</td>
<td>Length of time before next link-state transmission.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of LSA.</td>
</tr>
<tr>
<td>LS ID</td>
<td>Link-state ID of the LSA.</td>
</tr>
<tr>
<td>ADV RTR</td>
<td>IP address of advertising router.</td>
</tr>
<tr>
<td>Seq NO</td>
<td>Sequence number of LSA.</td>
</tr>
<tr>
<td>Age</td>
<td>Age of LSA (in seconds).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Checksum of LSA.</td>
</tr>
</tbody>
</table>
show ospfv3 interface

To display Open Shortest Path First Version 3 (OSPFv3) interface information, use the `show ospfv3 interface` command in XR EXEC mode.

```
show ospfv3 [process-name] [area-id] interface [type interface-path-id]
```

**Syntax Description**

- **process-name** (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.

- **area-id** (Optional) Area number used to define the particular area.

- **type** Interface type. For more information, use the `question mark (?)` online help function.

- **interface-path-id** Physical interface or virtual interface.

**Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the `question mark (?)` online help function.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

Release 6.0 This command was introduced.

**Usage Guidelines**

Use the `show ospfv3 interface` command when the adjacency between two neighboring routers is not forming. Adjacency means that the routers synchronize their databases when they discover each other.

You can look at the output to check the physical link and line protocol status and to confirm that the network type and timer intervals match those of the neighboring routers.

**Examples**

The following is sample output from the `show ospfv3 interface` command when tenGigE interface 0/2 5/0/0 is specified:

```
RP/0/RP0/CPU0:router# show ospfv3 interface GigabitEthernet 0/2
 5
```
tenGigE/25/0/0 is up, line protocol is up up
Link Local address fe80::203213:a0ff:1aff:fe9dfe14:f3fe3faa, Interface ID 2
Area 0, Process ID 1foo, Instance ID 0, Router ID 101.01.01.2061
Network Type BROADCAST, Cost: 101
BFD enabled, interval 300 msec, multiplier 5
Transmit Delay is 1 sec, State BDR, Priority 1
Designated Router (ID) 102.02.02.2072, local address
fe80::20421b:0ff:53ff:fe22fe74:73feeb6
Backup Designated router (ID) 101.01.01.2061, local address
fe80::203213:a0ff:1aff:fe9dfe14:f3fe3faa
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
Hello due in 00:00:06
Index 0/2/1, flood queue length 0
Last flood scan length is 21, maximum is 92
Last flood scan time is 0 msec, maximum is 100 msec
Neighbor Count is 1, Adjacent neighbor count is 1
Adjacent with neighbor 102.02.02.2072 (Designated Router)
Suppress hello for 0 neighbor(s)

This table describes the significant fields shown in the display.

Table 72: show ospfv3 interface Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tenGigE</td>
<td>Status of the physical link and operational status of the protocol.</td>
</tr>
<tr>
<td>Link Local Address</td>
<td>Interface link local address and interface ID.</td>
</tr>
<tr>
<td>Area</td>
<td>OSPFv3 area ID, process ID, instance ID, and router ID.</td>
</tr>
<tr>
<td>Transmit Delay</td>
<td>Transmit delay and interface state.</td>
</tr>
<tr>
<td>Designated Router</td>
<td>Designated router ID and respective interface IPv6 address.</td>
</tr>
<tr>
<td>Backup Designated router</td>
<td>Backup designated router ID and respective interface IPv6 address.</td>
</tr>
<tr>
<td>Timer intervals configured</td>
<td>Configuration of timer intervals.</td>
</tr>
<tr>
<td>Hello</td>
<td>Number of seconds until next hello packet is sent over this interface.</td>
</tr>
<tr>
<td>Index 0/2/1</td>
<td>Link, area and autonomous system flood indexes, and number of flood queue entries.</td>
</tr>
<tr>
<td>Next 0(0)/0(0)/0(0)</td>
<td>Next link, area and autonomous system flood information, data pointer, and index.</td>
</tr>
<tr>
<td>Last flood scan length</td>
<td>Length of last flood scan.</td>
</tr>
<tr>
<td>Last flood scan time</td>
<td>Time of last flood scan (in milliseconds).</td>
</tr>
<tr>
<td>Neighbor Count</td>
<td>Count of network neighbors and list of adjacent neighbors.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>Suppress hello</td>
<td>Count of neighbors suppressing hello messages.</td>
</tr>
</tbody>
</table>
**show ospfv3 message-queue**

To display the information about the queue dispatch values, peak lengths, and limits, use the `show ospfv3 message-queue` command in XR EXEC mode.

`show ospfv3 [process-name] message-queue`

This command has no keywords or arguments.

**Command Default**
None

**Command Modes**
XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

This is sample output from the `show ospfv3 message-queue` command:

```
RP/0/RP0/CPU0:router#show ospfv3 message-queue
Mon May 31 16:07:47.143 CEST

OSPFv3 Process 0
Hello Thread Packet Input Queue:
  Current queue length: 0
  Peak queue length: 2
  Queue limit: 5000
  Packets received: 104091
  Packets processed: 104091
  Packets dropped: 0
  Processing quantum: 10
  Full quantum used: 0
  Pulses sent: 104089
  Pulses received: 104089

Router Thread Message Queue
  Current queue length: 0
  Peak queue length: 2
  Low queue limit: 8000
  Medium queuing limit: 9000
  High queuing limit: 9500
  Messages queued: 1472
  Messages deleted: 0
  Messages processed: 1472
  Low queue drops: 0
  Medium queue drops: 0
  High queue drops: 0
```
show ospfv3 message-queue

Processing quantum: 300
Full quantum used: 0
Pulses sent: 1484
Pulses received: 1484
show ospfv3 neighbor

To display Open Shortest Path First Version 3 (OSPFv3) neighbor information on an individual interface basis, use the `show ospfv3 neighbor` command in XR EXEC mode.

```
show ospfv3 [process-name] [area-id] neighbor [type interface-path-id] [neighbor-id] [detail]
```

**Syntax Description**

- **process-name**: (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.

- **area-id**: (Optional) Area ID. If you do not specify an area, all areas are displayed.

- **type**: Interface type. For more information, use the question mark (?) online help function.

- **interface-path-id**: Physical interface or virtual interface.

  **Note**: Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

  For more information about the syntax for the router, use the question mark (?) online help function.

- **neighbor-id**: (Optional) Neighbor router ID.

- **detail**: (Optional) Displays all neighbors given in detail (lists all neighbors).

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show ospfv3 neighbor` command when the adjacency between two neighboring routers is not forming. Adjacency means that the routers synchronize their databases when they discover each other.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show ospfv3 neighbor` command showing two lines of summary information for each neighbor:

```
RP/0/RP0/CPU0:router# show ospfv3 neighbor

Neighbors for OSPFV3 1
```
This table describes the significant fields shown in the display.

**Table 73: show ospfv3 neighbor Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>Pri</td>
<td>Router priority for designated router election. A router with a priority of 0 is never elected as the designated router or backup designated router.</td>
</tr>
<tr>
<td>State</td>
<td>OSPFv3 state.</td>
</tr>
<tr>
<td>Dead Time</td>
<td>Time (in hh:mm:ss) to elapse before OSPFv3 declares the neighbor dead.</td>
</tr>
<tr>
<td>Interface ID</td>
<td>Number that uniquely identifies an interface on a router.</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the interface that connects to this neighbor.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Time (in hh:mm:ss) that the OSPFv3 neighbor has been up.</td>
</tr>
</tbody>
</table>

The following is sample output showing summary information about the neighbor that matches the neighbor ID:

```
RP/0/RP0/CPU0:router# show ospfv3 neighbor 10.0.0.207

Neighbors for OSPFv3 1

Neighbor 10.0.0.207
In the area 0 via interface Ethernet0/0/0/0
Neighbor: interface-id 2, link-local address fe80::204:c0ff:fe22:73fe
Neighbor priority is 1, State is FULL, 6 state changes
DR is 10.0.0.207 BDR is 10.0.0.206
Options is 0x13
Dead timer due in 00:00:38
Neighbor is up for 01:09:21
Index 0/1/2, retransmission queue length 0, number of retransmission 1
First 0(0)/0(0)/0(0) Next 0(0)/0(0)/0(0)
Last retransmission scan length is 1, maximum is 1
Last retransmission scan time is 0 msec, maximum is 0 msec

Neighbor 10.0.0.207
In the area 1 via interface tenGigE 0/3/0/0
Neighbor: interface-id 3, link-local address fe80::3034:30ff:fe33:3742
Neighbor priority is 1, State is FULL, 6 state changes
Options is 0x13
Dead timer due in 00:00:38
Neighbor is up for 01:09:21
```
This table describes the significant fields shown in the display.

**Table 74: show ospfv3 neighbor 10.0.0.207 Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor router ID.</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>Area and interface</td>
<td>Area and interface through which the OSPFv3 neighbor is known.</td>
</tr>
<tr>
<td>link-local address</td>
<td>Link local address of the interface.</td>
</tr>
<tr>
<td>Neighbor priority</td>
<td>Router priority of neighbor and neighbor state.</td>
</tr>
<tr>
<td>State</td>
<td>OSPFv3 state.</td>
</tr>
<tr>
<td>state changes</td>
<td>Number of state changes for this neighbor.</td>
</tr>
<tr>
<td>DR is</td>
<td>Neighbor ID of the designated router.</td>
</tr>
<tr>
<td>BDR is</td>
<td>Neighbor ID of the backup designated router.</td>
</tr>
<tr>
<td>Options</td>
<td>Hello packet options field contents (E-bit only; possible values are 0 and 2; 2 indicates that area is not a stub; 0 indi-</td>
</tr>
<tr>
<td>Dead timer</td>
<td>Time (in hh:mm:ss) to elapse before OSPFv3 declares the neighbor dead.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Time (in hh:mm:ss) that OSPFv3 neighbor has been up.</td>
</tr>
<tr>
<td>Index</td>
<td>Index and the remaining lines of this command give detailed information abou-</td>
</tr>
</tbody>
</table>

The following sample output shows the neighbors that match the neighbor ID on the interface when
the interface along with the neighbor ID is specified:

```
RP/0/RP0/CPU0:router# show ospfv3 neighbor tenGigE 0/3/0/1 10.0.0.207

Neighbors for OSPFv3 1

Neighbor 10.0.0.207
  In the area 0 via interface tenGigE 0/3/0/1
  Neighbor: interface-id 2, link-local address fe80::204:c0ff:fe22:73fe
  Neighbor priority is 1, State is FULL, 6 state changes
  DR is 10.0.0.207 BDR is 10.0.0.206
  Options is 0x13
  Dead timer due in 00:00:39
  Neighbor is up for 01:11:21
  Index 0/1/2, retransmission queue length 0, number of retransmission 1
```
This table describes the significant fields shown in the display.

Table 75: show ospfv3 neighbor tenGigE 0/3/0/1 10.0.0.207 Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>In the area</td>
<td>Area and interface through which the OSPFv3 neighbor is known.</td>
</tr>
<tr>
<td>link-local address</td>
<td>Link local address of the interface.</td>
</tr>
<tr>
<td>Neighbor priority</td>
<td>Router priority of neighbor and neighbor state.</td>
</tr>
<tr>
<td>State</td>
<td>OSPFv3 state.</td>
</tr>
<tr>
<td>state changes</td>
<td>Number of state changes for this neighbor.</td>
</tr>
<tr>
<td>DR is</td>
<td>Neighbor ID of the designated router.</td>
</tr>
<tr>
<td>BDR is</td>
<td>Neighbor ID of the backup designated router.</td>
</tr>
<tr>
<td>Options</td>
<td>Hello packet options field contents (E-bit only; possible values are 0 and 2; 2 indicates that area is not a stub; 0 indicates that area is a stub).</td>
</tr>
<tr>
<td>Dead timer</td>
<td>Time (in hh:mm:ss) to elapse before OSPFv3 declares the neighbor dead.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Time (in hh:mm:ss) that OSPFv3 neighbor has been up.</td>
</tr>
<tr>
<td>Index</td>
<td>Index and the remaining lines of this command give detailed information about flooding information received from the neighbor.</td>
</tr>
</tbody>
</table>

The following sample output shows all neighbors on the interface when the interface is specified:

```
RP/0/RP0/CPU0:router# show ospfv3 neighbor tenGigE 0/3/0/1
Neighbors for OSPFv3 1

Neighbor ID Pri State    Dead Time Interface ID  Interface
10.0.0.207   1    FULL/DR    00:00:37    2    tenGigE 0/3/0/1
Neighbor is up for 01:12:33

Total neighbor count: 1
```
This table describes the significant fields shown in the display.

**Table 76: show ospfv3 neighbor tenGigE 0/3/0/1 Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor ID</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>Pri</td>
<td>Router priority for designated router election. A router with a priority of 0 is never elected</td>
</tr>
<tr>
<td></td>
<td>as the designated router or backup designated router.</td>
</tr>
<tr>
<td>State</td>
<td>OSPF state.</td>
</tr>
<tr>
<td>Dead Time</td>
<td>Time (in hh:mm:ss) to elapse before OSPF declares the neighbor dead.</td>
</tr>
<tr>
<td>Interface ID</td>
<td>Number that uniquely identifies an interface on a router.</td>
</tr>
<tr>
<td>Interface</td>
<td>Name of the interface that connects to this neighbor.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Amount of time (in hh:mm:ss) that the OSPF neighbor has been up.</td>
</tr>
</tbody>
</table>

The following is sample output showing detailed neighbor information for tenGigE interface 0/3/0/1:

RP/0/RPO/CPU0:router# show ospfv3 neighbor tenGigE 0/3/0/1 detail

Neighbors for OSPFv3 1

Neighbor 10.0.0.207
In the area 0 via interface tenGigE 0/3/0/1
Neighbor: interface-id 2, link-local address fe80::204:c0ff:fe22:73fe
Neighbor priority is 1, State is FULL, 6 state changes
DR is 10.0.0.207 BDR is 10.0.0.206
Options is 0x13
Dead timer due in 00:00:39
Neighbor is up for 01:13:40
Index 0/1/2, retransmission queue length 0, number of retransmission 1
First 0(0)/0(0)/0(0) Next 0(0)/0(0)/0(0)
Last retransmission scan length is 1, maximum is 1
Last retransmission scan time is 0 msec, maximum is 0 msec

Total neighbor count: 1

This table describes the significant fields shown in the display.

**Table 77: show ospfv3 neighbor tenGigE 0/3/0/1 detail Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Neighbor router ID.</td>
</tr>
<tr>
<td>In the area</td>
<td>Area and interface through which the OSPFv3 neighbor is known.</td>
</tr>
<tr>
<td>link-local address</td>
<td>Link local address of the interface.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Neighbor priority</td>
<td>Router priority of neighbor and neighbor state.</td>
</tr>
<tr>
<td>State</td>
<td>OSPFv3 state.</td>
</tr>
<tr>
<td>state changes</td>
<td>Number of state changes for this neighbor.</td>
</tr>
<tr>
<td>DR is</td>
<td>Neighbor ID of the designated router.</td>
</tr>
<tr>
<td>BDR is</td>
<td>Neighbor ID of the backup designated router.</td>
</tr>
<tr>
<td>Options</td>
<td>Hello packet options field contents (E-bit only; possible values are 0 and 2; 2 indicates that area is not a stub; 0 indicates that area is a stub).</td>
</tr>
<tr>
<td>Dead timer</td>
<td>Time (in hh:mm:ss) to elapse before OSPFv3 declares the neighbor dead.</td>
</tr>
<tr>
<td>Neighbor is up</td>
<td>Time (in hh:mm:ss) that the OSPFv3 neighbor has been up.</td>
</tr>
<tr>
<td>Index</td>
<td>Index and the remaining lines of this command give detailed information about flooding information received from the neighbor.</td>
</tr>
</tbody>
</table>

The following is sample output showing neighbor information with BFD enabled for tenGigE interface 0/5/0/0:

```
RP/0/RP0/CPU0:router# show ospfv3 neighbor detail
Thu Sep 11 02:02:46.267 UTC

Neighbors for OSPFv3 foo

Neighbor 2.2.2.2
  In the area 0 via interface tenGigE 0/5/0/0   BFD enabled
  Neighbor: interface-id 2, link-local address fe80::21b:53ff:fe74:eab6
  Neighbor priority is 1, State is FULL, 6 state changes
  DR is 2.2.2.2 BDR is 1.1.1.1
  Options is 0x13
  Dead timer due in 00:00:32
  Neighbor is up for 00:06:16
  Index 1/1/1, retransmission queue length 0, number of retransmission 1
  First 0(0)/0(0)/0(0) Next 0(0)/0(0)/0(0)
  Last retransmission scan length is 1, maximum is 1
  Last retransmission scan time is 0 msec, maximum is 0 msec

Total neighbor count: 1
```
**show ospfv3 request-list**

To display the first ten link-state requests pending that the local router is making to the specified Open Shortest Path First Version 3 (OSPFv3) neighbor and interface, use the `show ospfv3 request-list` command in XR EXEC mode.

```plaintext
show ospfv3 [process-name] [area-id] request-list [type interface-path-id] [neighbor-id]
```

**Syntax Description**

- **process-name** (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.

- **area-id** (Optional) Area ID. If you do not specify an area, all areas are displayed.

- **type** (Optional) Interface type. For more information, use the question mark (?) online help function.

- **interface-path-id** (Optional) Physical interface or virtual interface.

  **Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

  For more information about the syntax for the router, use the question mark (?) online help function.

- **neighbor-id** (Optional) Router ID of the OSPFv3 neighbor. This argument must be in 32-bit dotted-decimal notation, similar to an IPv4 address.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You might use this command when the databases of two neighboring routers are out of synchronization or if the adjacency is not forming between them. Adjacency means that the routers synchronize their databases when they discover each other.

You can look at the list to determine if one router is trying to request a particular database update. Entries that appear to be suspended in the list usually indicate that updates are not being delivered. One possible reason for this behavior is a maximum transmission unit (MTU) mismatch between the routers.

You might also look at this list to make sure it is not corrupted. The list should refer to database entries that actually exist.
Examples

The following sample output shows request lists for neighbor 10.0.0.207 on the OSPFv3 1 process:

```
RP/0/RP0/CPU0:router# show ospfv3 1 request-list 10.0.0.207 tenGigE 0/3/0/0

Request Lists for OSPFv3 1

Neighbor 10.0.0.207, interface tenGigE 0/3/0/0 address fe80::3034:30ff:fe33:3742

<table>
<thead>
<tr>
<th>Type</th>
<th>LS ID</th>
<th>ADV RTR</th>
<th>Seq NO</th>
<th>Age</th>
<th>Checksum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>192.168.58.17</td>
<td>192.168.58.17</td>
<td>0x80000012</td>
<td>12</td>
<td>0x0036f3</td>
</tr>
<tr>
<td>2</td>
<td>192.168.58.68</td>
<td>192.168.58.17</td>
<td>0x80000012</td>
<td>12</td>
<td>0x00083f</td>
</tr>
</tbody>
</table>
```

This table describes the significant fields shown in the display.

**Table 78: show ospfv3 request-list Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Router ID of the neighboring router.</td>
</tr>
<tr>
<td>interface</td>
<td>Name of the interface that connects to this neighbor.</td>
</tr>
<tr>
<td>address</td>
<td>IPv6 address of the neighbor.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of link-state advertisement (LSA).</td>
</tr>
<tr>
<td>LS ID</td>
<td>Link-state ID of the LSA.</td>
</tr>
<tr>
<td>ADV RTR</td>
<td>Router ID of the advertising router.</td>
</tr>
<tr>
<td>Seq NO</td>
<td>Sequence number of the LSA.</td>
</tr>
<tr>
<td>Age</td>
<td>Age of the LSA (in seconds).</td>
</tr>
<tr>
<td>Checksum</td>
<td>Checksum of the LSA.</td>
</tr>
</tbody>
</table>
show ospfv3 retransmission-list

To display the first ten link-state entries in the retransmission list that the local router sends to the specified neighbor over the specified interface, use the `show ospfv3 retransmission-list` command in XR EXEC mode.

```
show ospfv3 [process-name ] [area-id] retransmission-list [type interface-path-id] [neighbor-id]
```

**Syntax Description**

- **process-name**: (Optional) Name that uniquely identifies an Open Shortest Path First Version 3 (OSPFv3) routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.

- **area-id**: (Optional) Area ID. If you do not specify an area, all areas are displayed.

- **type**: (Optional) Interface type. For more information, use the question mark (?) online help function.

- **interface-path-id**: (Optional) Physical interface or virtual interface.

  **Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

  For more information about the syntax for the router, use the question mark (?) online help function.

- **neighbor-id**: (Optional) IP address of the OSPFv3 neighbor.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You might use this command when the databases of two neighboring routers are out of synchronization or if the adjacency is not forming between them. Adjacency means that the routers synchronize their databases when they discover each other.

You can look at the list to determine if one router is trying to request a particular database update. Entries that appear to be suspended in the list usually indicate that updates are not being delivered. One possible reason for this behavior is a maximum transmission unit (MTU) mismatch between the routers.

You might also look at this list to make sure it is not corrupted. The list should refer to database entries that actually exist.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>
Examples

The following sample output shows the retransmission list for neighbor 10.0.124.4 on tenGigE interface 0/3/0/0:

```
RP/0/RP0/CPU0:router# show ospfv3 retransmission-list 10.0.124.4 tenGigE 0/3/0/0
Neighbor 10.0.124.4, interface tenGigE 0/3/0/0 address fe80::3034:30ff:fe33:3742
```

This table describes the significant fields shown in the display.

Table 79: show ospfv3 retransmission-list 10.0.124.4 tenGigE 0/3/0/0 Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor</td>
<td>Router ID of the neighboring router.</td>
</tr>
<tr>
<td>interface</td>
<td>Name of the interface that connects to this neighbor.</td>
</tr>
<tr>
<td>address</td>
<td>IPv6 address of the neighbor.</td>
</tr>
</tbody>
</table>
show ospfv3 routes

To display the Open Shortest Path First Version 3 (OSPFv3) route table, use the `show ospfv3 routes` command in XR EXEC mode.

```
show ospfv3 [process-name] routes [external|connected] [ipv6-prefix/prefix-length]
show ospfv3 [process-name] routes summary
```

**Syntax Description**

- **process-name** (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospf` command. If this argument is included, only information for the specified routing process is displayed.

- **external** (Optional) Displays routes redistributed from other protocols.

- **connected** (Optional) Displays connected routes.

- **ipv6-prefix** (Optional) IP Version 6 (IPv6) prefix, which limits output to a specific route. This argument must be in the form documented in RFC 2373, in which the address is specified in hexadecimal using 16-bit values between colons.

- **/prefix-length** (Optional) Length of the IPv6 prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash must precede the decimal value.

- **summary** Displays a summary of the route table.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

**Release** Modification

Release 6.0  This command was introduced.

**Usage Guidelines**

Use the `show ospfv3 routes` command to display the OSPFv3 private routing table (which contains only routes calculated by OSPFv3). If there is something wrong with a route in the Routing Information Base (RIB), check the OSPFv3 copy of the route to determine if it matches the RIB contents. If it does not match, there is a synchronization problem between OSPFv3 and the RIB. If the routes match and the route is incorrect, OSPFv3 has made an error in its routing calculation.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following sample output shows the route table for OSPFv3 process 1:
show ospfv3 1 routes

Route Table for OSPFV3 1 with ID 10.3.4.2

* 3000:11:22::/64, Inter, cost 21/0, area 1
tenGigE 0/3/0/0, fe80::3034:30ff:fe33:3742
10.0.0.207/200
* 3000:11:22:1::/64, Inter, cost 31/0, area 1
tenGigE 0/3/0/0, fe80::3034:30ff:fe33:3742
10.0.0.207/1
* 3333::/56, Ext2, cost 20/1, P:0 F:0
tenGigE 0/3/0/0, fe80::3034:30ff:fe33:3742
10.0.0.207/0
* 6050::/56, Ext2, cost 20/1, P:0 F:0
tenGigE 0/3/0/0, fe80::3034:30ff:fe33:3742
10.0.0.207/1
* 7002::/56, Intra, cost 10/0, area 0
tenGigE 0/0/0/0, connected
* 3000:11:22::/64, Inter, cost 21/0, area 1
tenGigE 0/3/0/0, fe80::3034:30ff:fe33:3742
10.0.0.207/200

This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3000:11:22::/64</td>
<td>Route prefix to the local router.</td>
</tr>
<tr>
<td>Inter</td>
<td>Prefix 3000:11:22::/64 is interarea.</td>
</tr>
<tr>
<td>cost 21/0</td>
<td>Sum of the link costs required to reach prefix 3000:11:22::/64. 0. In this example, 20 is the external cost.</td>
</tr>
<tr>
<td>tenGigE 0/3/0/0</td>
<td>Packets destined for prefix 3000:11:22::/64 are sent over the tenGigE 0/3/0/0 interface.</td>
</tr>
<tr>
<td>fe80::3034:30ff:fe33:3742</td>
<td>Next-hop router on the path to prefix 3000:11:22::/64.</td>
</tr>
<tr>
<td>10.0.0.207</td>
<td>Router 10.0.0.207 is the router that advertised this route.</td>
</tr>
</tbody>
</table>
show ospfv3 statistics rib-thread

To display RIB thread statistics, use the **show ospfv3 statistics rib-thread** command in XR EXEC mode.

```
show ospfv3 [process-name [area-id]] statistics rib-thread
```

**Syntax Description**

- **process-name** (Optional) Name that uniquely identifies an OSPF routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.

- **area-id** (Optional) Area number used to define the particular area.

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

This is sample output from **show ospfv3 statistics rib-thread** command:

```
RP/0/RP0/CPU0:router#show ospfv3 0 statistics rib-thread
Mon May 10 17:48:29.011 CEST
OSPFv3 0 RIB thread statistics
Queue statistics:
 Last entry dequeue 10127056 msecs ago (14:59:42.171)
 RIB thread active NO
 Total RIB thread signals 30
 Current queue length 0
 Maximum queue length 2
 Total entries queued 31
 Total entries dequeued 31
 Maximum latency (msec) 5.000
 Average latency (msec) 0.323
 Queue errors:
 Enqueue errors 0
 Dequeue errors 0
 RIB batch statistics:
 Batches sent to RIB 31
 Batch all routes OK 31
 Batch some routes backup 0
 RIB batch errors:
 Batches version mismatch 0
 Batches missing connection 0
```
<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batches no table</td>
<td>0</td>
</tr>
<tr>
<td>Batch route table limit</td>
<td>0</td>
</tr>
<tr>
<td>Batch route errors</td>
<td>0</td>
</tr>
<tr>
<td>Batch errors</td>
<td>0</td>
</tr>
<tr>
<td>Route table limit</td>
<td>0</td>
</tr>
<tr>
<td>Route path errors</td>
<td>0</td>
</tr>
<tr>
<td>Route errors</td>
<td>0</td>
</tr>
<tr>
<td>Path table limit</td>
<td>0</td>
</tr>
<tr>
<td>Path errors</td>
<td>0</td>
</tr>
</tbody>
</table>
**show ospfv3 summary-prefix**

To display Open Shortest Path First Version 3 (OSPFv3) aggregated summary address information, use the `show ospfv3 summary-prefix` command in XR EXEC mode.

```
show ospfv3 [process-name] summary-prefix
```

### Syntax Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>process-name</td>
<td>(Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the <code>router ospfv3</code> command. If this argument is included, only information for the specified routing process is displayed.</td>
</tr>
</tbody>
</table>

### Command Default

No default behavior or values

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `show ospfv3 summary-prefix` command if you configured summarization of external routes with the `summary-prefix` command and you want to display configured summary addresses.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

### Examples

The following sample output shows the summary prefix address for the OSPFv3 1 process:

```
RP/0/RP0/CPU0:router# show ospfv3 1 summary-prefix
OSPFv3 Process 1, Summary-prefix
  4004:f000::/32 Metric 20, Type 2, Tag 0
```

This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>4004:f000::/32</td>
<td>Summary prefix designated for a range of IPv6 prefixes. The length of the IPv6 prefix.</td>
</tr>
<tr>
<td>Metric</td>
<td>Metric used to advertise the summary routes.</td>
</tr>
<tr>
<td>Type</td>
<td>External link-state advertisements (LSAs) metric type.</td>
</tr>
</tbody>
</table>
### Field | Description
---|---
Tag | Tag value that can be used as a “match” value for controlling redistribution through route maps.
show ospfv3 virtual-links

To display parameters and the current state of Open Shortest Path First Version 3 (OSPFv3) virtual links, use the `show ospfv3 virtual-links` command in XR EXEC mode.

```
show ospfv3 [process-name] virtual-links
```

**Syntax Description**

`process-name`  (Optional) Name that uniquely identifies an OSPFv3 routing process. The process name is defined by the `router ospfv3` command. If this argument is included, only information for the specified routing process is displayed.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The information displayed in the `show ospfv3 virtual-links` command is useful in debugging OSPFv3 routing operations.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following sample output shows the virtual links for the OSPFv3 1 process:

```
RP/0/RP0/CPU0:router# show ospfv3 1 virtual-links

Virtual Links for OSPFv3 1
Virtual Link to router 172.31.101.2 is up
  Interface ID 16, IPv6 address 3002::206
  Transit area 0.0.0.1, via interface tenGigE 0/3/0/0, Cost of using 11
  Transmit Delay is 5 sec, State POINT_TO_POINT,
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
  Hello due in 0:00:08
  Adjacency State FULL

This table describes the significant fields shown in the display.

*Table 82: show ospfv3 virtual-links Field Descriptions*

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Virtual Link to router is up</td>
<td>Specifies the OSPFv3 neighbor, and if the link to that neighbor is up or down.</td>
</tr>
</tbody>
</table>
### Field | Description
--- | ---
Interface ID | ID of the virtual link interface.
IPv6 address | IPv6 address of virtual link endpoint.
Transit area | Transit area through which the virtual link is formed.
via interface | Interface through which the virtual link is formed.
Cost | Cost of reaching the OSPF neighbor through the virtual link.
Transmit Delay | Transmit delay on the virtual link.
State POINT_TO_POINT | State of the OSPFv3 neighbor.
Timer intervals | Various timer intervals configured for the link.
Hello due in | When the next hello message is expected from the neighbor (in hh:mm:ss).
Adjacency State | Adjacency state between the neighbors.
show protocols (OSPFv3)

To display information about the Open Shortest Path First Version 3 (OSPFv3) process running on the router, use the show protocols command in XR EXEC mode.

```
show protocols [{afi-all|ipv4|ipv6}] [{all|protocol}]
```

**Syntax Description**
- `afi-all` (Optional) Specifies all address families.
- `ipv4` (Optional) Specifies an IPv4 address family.
- `ipv6` (Optional) Specifies an IPv6 address family.
- `all` (Optional) Specifies all protocols for a given address family.
- `protocol` (Optional) Specifies a routing protocol. For the IPv4 address family, the options are:
  - `bgp`
  - `isis`
  - `ospf`
  - `rip`
  
  For the IPv6 address family, the options are:
  - `bgp`
  - `isis`
  - `ospfv3`

**Command Default**
The default address family is IPv4.

**Command Modes**
XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**
The following example is sample output from the show protocols command:

```
RP/0/RP0/CPU0:router# show protocols ipv6 ospfv3
Routing Protocol OSPFv3 1
  Router Id:10.0.0.1
```
Distance: 110
Redistribution: None
Area 0
tenGigE 0/2/0/2
Loopback1

This table describes the significant fields shown in the display.

Table 83: show protocols Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router Id</td>
<td>Router ID of the OSPFv3 process.</td>
</tr>
<tr>
<td>Distance</td>
<td>Administrative distance for the protocol. This distance determines the priority the Routing Information Base (RIB) gives to the routes, as opposed to other protocols, for example, IS-IS.</td>
</tr>
<tr>
<td>Redistribution</td>
<td>Protocols from which this OSPFv3 process is redistributing routes.</td>
</tr>
<tr>
<td>Area</td>
<td>OSPFv3 areas defined in this process, followed by their associated interfaces.</td>
</tr>
</tbody>
</table>
snmp context (OSPFv3)

To specify an SNMP context for an OSPFv3 instance, use the `snmp context` command in router configuration mode or in VRF configuration mode. To remove the SNMP context, use the `no` form of this command.

```
snmp context context_name
no snmp context context_name
```

**Syntax Description**

- `context_name` Specifies name of the SNMP context for OSPFv3 instance.

**Command Default**

SNMP context is not specified.

**Command Modes**

- Router OSPFv3 configuration
- VRF configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

To use this command, you must be in a user group associated with a task group that includes the proper task IDs. If you suspect user group assignment is preventing you from using a command, contact your AAA administrator for assistance.

The snmp-server commands need to be configured to perform SNMP request for the OSPF instance. Refer `SNMP Server Commands` module in System Management Command Reference for information on using the snmp-server commands.

**Note**

To map an SNMP context with a protocol instance, topology or VRF entity, use the `snmp-server context mapping` command. However, the `feature` option of this command does not work with OSPFv3 protocol.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
<td></td>
</tr>
</tbody>
</table>

This example shows how to configure an SNMP context `foo` for OSPFv3 instance `100`:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospfv3 100
RP/0/RP0/CPU0:router(config-ospf)#snmp context foo
```

This example shows how to configure `snmp-server` commands to be used with the `snmp context` command:
RP/0/RP0/CPU0:router(config)#snmp-server host 10.0.0.2 traps version 2c public udp-port 1620
RP/0/RP0/CPU0:router(config)#snmp-server community public RW
RP/0/RP0/CPU0:router(config)#snmp-server contact foo
RP/0/RP0/CPU0:router(config)#snmp-server community-map public context foo

This is a sample SNMP context configuration for OSPFv3 instance 100:

```
snmp-server host 10.0.0.2 traps version 2c public udp-port 1620
snmp-server community public RW
snmp-server contact foo
snmp-server community-map public context foo

router ospfv3 100
  router-id 2.2.2.2
  bfd fast-detect
  nsf cisco
  snmp context foo
  area 0
    interface Loopback1
      !
  area 1
    interface tenGigE 0/2/0/1
      demand-circuit enable
      !
    interface POS0/3/0/0
      !
    interface POS0/3/0/1
      !
```

Routing Command Reference for Cisco NCS 5000 Series Routers
snmp trap (OSPFv3)

To enable SNMP trap for an OSPFv3 instance, use the `snmp trap` command in VRF configuration mode. To disable SNMP trap for the OSPFv3 instance, use the `no` form of this command.

```
snmp trap
no snmp trap
```

**Syntax Description**

This command has no keywords or arguments.

**Command Default**

Disabled.

**Command Modes**

OSPFv3 VRF configuration

**Command History**

Release | Modification
---|---
6.0 | This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to enable SNMP trap for OSPFv3 instance 100 under VRF `vrf-1`:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospfv3 100
RP/0/RP0/CPU0:router(config-ospf)#vrf vrf-1
RP/0/RP0/CPU0:router(config-ospf-vrf)#snmp trap
```
snmp trap rate-limit (OSPFv3)

To control the number of traps that OSPFv3 sends by configuring window size and the maximum number of traps during that window, use the `snmp trap rate-limit` command in router OSPFv3 configuration mode or OSPFv3 VRF configuration mode. To disable configuring the window size and maximum number of traps during the window, use the `no` form of this command.

```
snmp trap rate-limit window-size max-num-traps
no snmp trap rate-limit window-size max-num-traps
```

**Syntax Description**

- `window-size` Specifies the trap rate limit sliding window size. The range is 2 to 60 windows.
- `max-num-traps` Specifies the maximum number of traps sent in window time. The range is 0 to 300 traps.

**Command Default**

None

**Command Modes**

Router OSPFv3 configuration

OSPFv3 VRF configuration

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to set the trap rate limit sliding window size to 50 and the maximum number of traps sent to 250 for OSPFv3 instance 100 under vrf vrf1:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospfv3 100
RP/0/RP0/CPU0:router(config-ospfv3)#vrf vrf1
RP/0/RP0/CPU0:router(config-ospfv3-vrf)#snmp trap rate-limit 50 250
```
spf prefix-priority (OSPFv3)

To prioritize OSPFv3 prefix installation into the global Routing Information Base (RIB) during Shortest Path First (SPF) run, use the `spf prefix-priority` command in XR Config mode or VRF configuration mode. To return to the system default value, use the `no` form of this command.

```
spf prefix-priority route-policy policy-name [disable]
spf prefix-priority route-policy policy-name
```

**Syntax Description**

- **route-policy** Specifies the route-policy to prioritize route installation.
- **policy-name** Name of the route policy.
- **disable** Disables SPF prefix priority

**Command Default**

SPF prefix prioritization is disabled.

**Command Modes**

- XR Config mode
- VRF configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read,</td>
</tr>
<tr>
<td></td>
<td>write</td>
</tr>
</tbody>
</table>

This example shows how to configure OSPFv3 SPF prefix prioritization:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# prefix-set ospf3-critical-prefixes
RP/0/RP0/CPU0:router(config-pfx)# 66.0.0.0/16
RP/0/RP0/CPU0:router(config-pfx)# end-set
RP/0/RP0/CPU0:router(config)# route-policy ospf3-spf-priority
RP/0/RP0/CPU0:router(config-rpl)# if destination in ospf-critical-prefixes then set spf-priority critical
endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
RP/0/RP0/CPU0:router(config-rpl)# commit
RP/0/RP0/CPU0:router(config-rpl)# exit
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospf)# router-id 66.0.0.1
RP/0/RP0/CPU0:router(config-ospf)# spf prefix-priority route-policy ospf-spf-priority
```
To define an area as a stub area for Open Shortest Path First Version 3 (OSPFv3), use the `stub` command in area configuration mode. To disable this function, use the `no` form of this command.

```
stub [no-summary]
no stub
```

**Syntax Description**

- **no-summary** (Optional) Prevents an area border router (ABR) from sending summary link advertisements into the stub area. Areas with this option are known as **totally stubby areas**.

**Command Default**

No stub area is defined.

**Command Modes**

Area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

You must configure the `stub` command on all routers in the stub area. Use the `default-cost area` command on the ABR of a stub area to specify the cost of the default route advertised into the stub area by the ABR.

Two stub area router configuration commands exist: the `stub` and `default-cost` commands. In all routers attached to the stub area, the area should be configured as a stub area using the `stub` command. Use the `default-cost` command only on an ABR attached to the stub area. The `default-cost` command provides the metric for the summary default route generated by the ABR into the stub area.

To further reduce the number of link-state advertisements (LSAs) sent into a stub area, you can configure the `no-summary` keyword on the ABR to prevent it from sending summary LSAs (LSA Type 3) into the stub area.

A stub area does not accept information about routes external to the autonomous system.

**Examples**

The following example shows how to create stub area 5 and specifies a cost of 20 for the default summary route sent into this stub area:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 201
RP/0/RP0/CPU0:router(config-ospfv3)# area 5
RP/0/RP0/CPU0:router(config-ospfv3-ar)# stub
RP/0/RP0/CPU0:router(config-ospfv3-ar)# default-cost 20
```
stub-router

To modify self originated router LSAs when stub router is active, use the **stub-router** command in an appropriate configuration mode. To disable this function, use the **no** form of this command.

```
stub-router router-lsa [{r-bit|v6-bit|max-metric}] [always] [on-proc-migration interval]
on-proc-restart interval [on-switchover interval] [on-startup [{interval]wait-for-bgp}] [summary-lsa
metric] [external-lsa [metric]] [include-stub]
stub-router router-lsa [{r-bit|v6-bit|max-metric}]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>router-lsa</td>
<td>Specifies that always originate router link-state advertisements (LSAs) with the stub-router.</td>
</tr>
<tr>
<td>r-bit</td>
<td>Router-LSAs are originated with R-bit clear (v6 bit set), which means the node does not act as a transit router. Directly connected networks (native to OSPF) are still reachable within the OSPF area.</td>
</tr>
<tr>
<td>v6-bit</td>
<td>Router-LSAs are originated with V6 bit clear (and also r-bit clear). That means the node is not willing to receive any ipv6 traffic. Other ospfv3 routers won't install any route to a node with v6-bit clear.</td>
</tr>
<tr>
<td>max-metric</td>
<td>Router-LSAs are originated with maximum metric. Unlike the r-bit and v6-bit mode, the router may still act as a transit node, if there is no alternate path.</td>
</tr>
<tr>
<td>always</td>
<td>Stub-router mode is activated unconditionally.</td>
</tr>
<tr>
<td>on-proc-migration</td>
<td>Stub-router mode is activated for the desired period of time, upon ospfv3 process migration.</td>
</tr>
<tr>
<td>on-proc-restart</td>
<td>Stub-router mode is activated for the desired period of time, upon ospfv3 process restart.</td>
</tr>
<tr>
<td>on-switchover</td>
<td>Stub-router mode is activated for the desired period of time, upon RP failover.</td>
</tr>
<tr>
<td>on-startup</td>
<td>Stub-router mode is activated (for configured time, or until BGP converges) upon router startup (boot).</td>
</tr>
<tr>
<td>wait-for-bgp</td>
<td>Stub-router mode is terminated upon BGP convergence in ipv6 unicast address family. This option could only be used in the global routing table, not in a non-default VRF. This option is only supported with the on-startup trigger when the router boots.</td>
</tr>
<tr>
<td>summary-lsa</td>
<td>If enabled, summary LSAs are advertised with modified metric when stub-router is active. This configuration is applicable to max-metric mode.</td>
</tr>
<tr>
<td></td>
<td>In r-bit mode, ABR/ASBR functionality is implicitly disabled and routers will not use this node as an ABR/ASBR, since it declares no transit capability (r-bit clear).</td>
</tr>
<tr>
<td></td>
<td>If enabled and metric is not explicitly configured, the default metric for summary LSAs when stub-router active is 16711680 (0xFF0000).</td>
</tr>
</tbody>
</table>
### external-lsa

If enabled, external LSAs are advertised with modified metric when stub-router is active. This configuration is applicable to max-metric mode.

In r-bit mode, ABR/ASBR functionality is implicitly disabled and routers will not use this node as an ABR/ASBR, since it declares no transit capability (r-bit clear).

If enabled and metric is not explicitly configured, the default metric for external LSAs when stub-router active is 16711680 (0xFF0000).

### include-stub

If enabled, intra-area-prefix LSAs that are referencing router LSA are advertised with maximum metric (0xffff) when stub-router is active.

Intra-area-prefix LSAs that are referencing network LSA do not change metric.

Can be used in r-bit and max-metric modes.

/128 prefixes that are normally advertised with LA-bit set and 0 metric are also advertised with maximum metric and LA-bit clear when stub-router is active.

---

**Command Default**

Disabled.

**Command Modes**

Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Only one method (r-bit, v6-bit, max-metric) could be activated at a time. Configuring the methods simultaneously, or different method per trigger, is not supported.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to configure router LSAs are originated with R-bit clear under OSPFv3 VRF, `vrf_1`:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)#vrf vrf_1
RP/0/RP0/CPU0:router(config-ospfv3-vrf)#stub-router router-lsa r-bit
```
**summary-prefix (OSPFv3)**

To create aggregate addresses for routes being redistributed from another routing protocol into Open Shortest Path First Version 3 (OSPFv3) protocol, use the `summary-prefix` command in an appropriate configuration mode. To stop summarizing redistributed routes, use the `no` form of the command.

```
summary-prefix ipv6-prefix/prefix-length [not-advertise] tag tag
no summary-prefix ipv6-prefix/prefix-length
```

**Syntax Description**

- `ipv6-prefix` Summary prefix designated for a range of IP Version 6 (IPv6) prefixes. This argument must be in the form documented in RFC 2373, where the address is specified in hexadecimal using 16-bit values between colons.
- `/ prefix-length` Length of the IPv6 prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash must precede the decimal value.
- `not-advertise` (Optional) Suppresses summary routes that match the address and mask pair from being advertised.
- `tag` (Optional) Specifies a tag value that can be used as a “match” value for controlling redistribution.

**Command Default**

When this command is not used in router configuration mode, aggregate addresses are not created for routes being redistributed from another routing protocol into the OSPFv3 protocol.

**Command Modes**

- Router OSPFv3 configuration
- OSPFv3 VRF configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `summary-prefix` command to cause an OSPFv3 Autonomous System Boundary Router (ASBR) to advertise one external route as an aggregate for all redistributed routes that are covered by the address. This command summarizes only routes from other routing protocols that are being redistributed into OSPFv3.

You can use this command multiple times to summarize multiple groups of addresses. The metric used to advertise the summary is the lowest metric of all the more specific routes. This command helps reduce the size of the routing table.

If you want to summarize routes between OSPFv3 areas, use the `range` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>
In the following example, if summary prefix 4004:f000:132 is configured and routes 4004:f000:1::/64, 4004:f000:2::/64, and 4004:f000:3::/64 are redistributed into OSPFv3; only route 4004:f000::/32 is advertised in an external link-state advertisement:

```
RP/0/RP0/CPU0:router(config-ospfv3)# summary-prefix 4004:f000::/32
```
**timers lsa arrival**

To set the minimum interval at which the software accepts the same link-state advertisement (LSA) from Open Shortest Path First Version 3 (OSPFv3) neighbors, use the `timers lsa arrival` command in an appropriate configuration mode. To restore the default value, use the `no` form of this command.

```
timers lsa arrival milliseconds
no timers lsa arrival
```

**Syntax Description**

- **milliseconds**
  
  Minimum delay (in milliseconds) that must pass between acceptance of the same LSA arriving from neighbors. Range is 0 to 60000 milliseconds.

**Command Default**

1000 milliseconds

**Command Modes**

Router OSPFv3 configuration

**Command History**

- **Release** Modification
  - Release 6.0  This command was introduced.

**Usage Guidelines**

Use the `timers lsa arrival` command to control the minimum interval for accepting the same LSA. The same LSA is an LSA instance that contains the same LSA ID number, LSA type, and advertising router ID. If an instance of the same LSA arrives sooner than the interval that is set, the LSA is dropped.

We recommend that the `milliseconds` value of the `timers lsa arrival` command be less than or equal to the `hold-interval` value of the `timers throttle lsa all` command for the neighbor.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the minimum interval for accepting the same LSA at 2000 milliseconds:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# timers throttle lsa all 200 10000 45000
RP/0/RP0/CPU0:router(config-ospfv3)# timers lsa arrival 2000
```
timers pacing flood

To configure link-state advertisement (LSA) flood packet pacing, use the `timers pacing flood` command in an appropriate configuration mode. To restore the default flood packet pacing value, use the `no` form of this command.

```
timers pacing flood milliseconds
no timers pacing flood
```

**Syntax Description**

- **milliseconds**: Time (in milliseconds) at which LSAs in the flooding queue are paced in between updates. Range is 5 milliseconds to 100 milliseconds.

**Command Default**

- **milliseconds**: 33

**Command Modes**

- Router OSPFv3 configuration

**Command History**

- **Release 6.0**: This command was introduced.

**Usage Guidelines**

Configuring OSPFv3 flood pacing timers allows you to control interpacket spacing between consecutive link-state update packets in the OSPF transmission queue. Use the `timers pacing flood` command to control the rate at which LSA updates occur, thereby preventing high CPU or buffer utilization that can result when an area is flooded with a very large number of LSAs.

The default settings for OSPFv3 packet pacing timers are suitable for the majority of OSPFv3 deployments. Do not change the packet pacing timers unless all other options to meet OSPFv3 packet flooding requirements have been exhausted. Specifically, network operators should prefer summarization, stub area usage, queue tuning, and buffer tuning before changing the default flood timers. Furthermore, no guidelines exist for changing timer values; each OSPFv3 deployment is unique and should be considered on a case-by-case basis. The network operator assumes risks associated with changing the default flood timer values.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure LSA flood packet-pacing updates to occur in 55-millisecond intervals for OSPFv3 routing process 1:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# timers pacing flood 55
```
timers pacing lsa-group

To change the interval at which Open Shortest Path First Version 3 (OSPFv3) link-state advertisements (LSAs) are collected into a group and refreshed, checksummed, or aged, use the `timers pacing lsa-group` command in an appropriate configuration mode. To restore the default value, use the `no` form of this command.

```
timers pacing lsa-group seconds
no timers pacing lsa-group
```

**Syntax Description**

| `seconds` | Interval (in seconds) at which LSAs are grouped and refreshed, checksummed, or aged. Range is 10 to 1800 seconds. |

**Command Default**

```
seconds : 240
```

OSPFv3 LSA group pacing is enabled by default.

**Command Modes**

Router OSPFv3 configuration

**Command History**

```
Release Modification
Release 6.0    This command was introduced.
```

**Usage Guidelines**

Use the `timers pacing lsa-group` command to control the rate at which LSA updates occur so that high CPU or buffer utilization that can occur when an area is flooded with a very large number of LSAs can be reduced. The default settings for OSPFv3 packet pacing timers are suitable for the majority of deployments. Do not change the packet pacing timers unless all other options to meet OSPFv3 packet flooding requirements have been exhausted. Specifically, network operators should prefer summarization, stub area usage, queue tuning, and buffer tuning before changing the default flooding timers. Furthermore, no guidelines exist for changing timer values; each OSPFv3 deployment is unique and should be considered on a case-by-case basis. The network operator assumes the risks associated with changing the default timer values.

Cisco IOS XR software groups the periodic refresh of LSAs to improve the LSA packing density for the refreshes in large topologies. The group timer controls the interval used for group refreshment of LSAs; however, this timer does not change the frequency that individual LSAs are refreshed (the default refresh rate is every 30 minutes).

The duration of the LSA group pacing is inversely proportional to the number of LSAs the router is handling. For example, if you have about 10,000 LSAs, decreasing the pacing interval would benefit you. If you have a very small database (40 to 100 LSAs), increasing the pacing interval to 10 to 20 minutes might benefit you slightly.

**Task ID**

```
Task ID Operations
ospf    read, write
```

**Examples**

The following example shows how to configure OSPFv3 group packet-pacing updates between LSA groups to occur in 60-second intervals for OSPFv3 routing process 1:

```
Routing Command Reference for Cisco NCS 5000 Series Routers 453
```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# timers pacing lsa-group 60


timers pacing retransmission

To configure link-state advertisement (LSA) retransmission packet pacing, use the `timers pacing retransmission` command in an appropriate configuration mode. To restore the default retransmission packet pacing value, use the `no` form of this command.

```
timers pacing retransmission milliseconds
no timers pacing retransmission
```

**Syntax Description**

- `milliseconds`: Time (in milliseconds) at which LSAs in the retransmission queue are paced. Range is 5 milliseconds to 100 milliseconds.

**Command Default**

`milliseconds`: 66

**Command Modes**

Router OSPFv3 configuration

**Command History**

- **Release Modification**
  - Release 6.0  This command was introduced.

**Usage Guidelines**

Use the `timers pacing retransmission` command to control interpacket spacing between consecutive link-state update packets in the OSPFv3 retransmission queue. This command controls the rate at which LSA updates occur. When an area is flooded with a very large number of LSAs, the LSA updates can result in high CPU or buffer utilization. Using this command reduces CPU or buffer utilization.

The default settings for OSPFv3 packet retransmission pacing timers are suitable for the majority of deployments. Do not change the packet retransmission pacing timers unless all other options to meet OSPFv3 packet flooding requirements have been exhausted. Specifically, network operators should prefer summarization, stub area usage, queue tuning, and buffer tuning before changing the default flooding timers. Furthermore, no guidelines exist for changing timer values; each OSPFv3 deployment is unique and should be considered on a case-by-case basis. The network operator assumes risks associated with changing the default packet retransmission pacing timer values.

**Task ID**

- `Task ID` `Operations`
  - ospf  read, write

**Examples**

The following example shows how to configure LSA flood pacing updates to occur in 55-millisecond intervals for OSPFv3 routing process 1:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# timers pacing retransmission 55
```
timers throttle lsa all (OSPFv3)

To set rate-limiting values for Open Shortest Path First Version 3 (OSPFv3) link-state advertisement (LSA) generation, use the `timers throttle lsa all` command in an appropriate configuration mode. To restore the default values, use the `no` form of this command.

```
timers throttle lsa all start-interval hold-interval max-interval
```

```
no timers throttle lsa all
```

**Syntax Description**

- `start-interval` Minimum delay (in milliseconds) for the generation of LSAs. The first instance of LSA is always generated immediately upon a local OSPFv3 topology change. The generation of the next LSA is not before the start interval. Range is 0 to 600000 milliseconds.

- `hold-interval` Incremental time (in milliseconds). This value is used to calculate the subsequent rate limiting times for LSA generation. Range is 1 to 600000 milliseconds.

- `max-interval` Maximum wait time (in milliseconds) between generation of the same LSA. Range is 1 to 600000 milliseconds.

**Command Default**

- `start-interval`: 500 milliseconds 50 milliseconds
- `hold-interval`: 5000 milliseconds 200 milliseconds
- `max-interval`: 5000 milliseconds

**Command Modes**

Router OSPFv3 configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The “same LSA” is defined as an LSA instance that contains the same LSA ID number, LSA type, and advertising router ID. We recommend that you keep the `milliseconds` value of the `timers lsa arrival` command less than or equal to the `hold-interval` value of the `timers throttle lsa all` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

This example shows how to customize OSPFv3 LSA throttling so that the start interval is 200 milliseconds, the hold interval is 10,000 milliseconds, and the maximum interval is 45,000 milliseconds. The minimum interval between instances of receiving the same LSA is 2000 milliseconds.

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# timers throttle lsa all 200 10000 45000
```
RP/0/RP0/CPU0:router(config-ospfv3)# timers lsa arrival 2000
timers throttle spf (OSPFv3)

To turn on Open Shortest Path First Version 3 (OSPFv3) shortest path first (SPF) throttling, use the **timers throttle spf** command in an appropriate configuration mode. To turn off SPF throttling, use the **no** form of this command.

```
timers throttle spf spf-start spf-hold spf-max-wait
no timers throttle spf spf-start spf-hold spf-max-wait
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>spf-start</td>
<td>Initial SPF schedule delay (in milliseconds). Range is 1 to 600000 milliseconds.</td>
</tr>
<tr>
<td>spf-hold</td>
<td>Minimum hold time (in milliseconds) between two consecutive SPF calculations. Range is 1 to 600000 milliseconds.</td>
</tr>
<tr>
<td>spf-max-wait</td>
<td>Maximum wait time (in milliseconds) between two consecutive SPF calculations. Range is 1 to 600000 milliseconds.</td>
</tr>
</tbody>
</table>

### Command Default

- **spf-start**: 5000 milliseconds
- **spf-hold**: 10000 milliseconds
- **spf-max-wait**: 10000 milliseconds

### Command Modes

Router OSPFv3 configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

The first wait interval between SPF calculations is the amount of time (in milliseconds) specified by the `spf-start` argument. Each consecutive wait interval is twice the current hold level (in milliseconds) until the wait time reaches the maximum time (in milliseconds) as specified by the `spf-max-wait` argument. Subsequent wait times remain at the maximum until the values are reset or a link-state advertisement (LSA) is received between SPF calculations.

### Tip

Setting a low `spf-start` time and `spf-hold` time causes routing to switch to the alternate path more quickly if a failure occurs. However, it consumes more CPU processing time.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to change the start, hold, and maximum wait interval values to 5, 1000, and 90,000 milliseconds, respectively:
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# timers throttle spf 5 1000 90000
trace (OSPFv3)

To specify the Open Shortest Path First Version 3 (OSPFv3) buffer size, use the `trace` command in router ospfv3 configuration mode. To return to the default value, use the `no` form of this command.

```
trace size buffer_name size
no trace size buffer_name size
```

**Syntax Description**

- `size` Deletes existing buffer and creates one with \( N \) entries.
- `buffer_name` Specifies a buffer from one of the 15 listed buffers. Refer Table 84: Buffer Types, on page 460 table for details on the buffers.
- `size` Specifies allowed size for the selected buffer. Options are: 0, 256, 1024, 2048, 4096, 8192, 16384, 32768, and 65536.
  
  Select 0 to disable traces.

**Command Default**

No default behavior or values

**Command Modes**

Router ospfv3 configuration

**Command History**

Release 6.0 This command was introduced.

**Usage Guidelines**

Trace buffers are used to store various traffic and processing events during the runtime. Large buffers can store more events. If the buffer becomes full, old entries are overwritten by the latest entries. In a large network, user may want to increase the trace buffer size to accommodate more events.

**Table 84: Buffer Types**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>adj</td>
<td>adjacency</td>
</tr>
<tr>
<td>adj_cycle</td>
<td>dbd/flood events/pkts</td>
</tr>
<tr>
<td>config</td>
<td>config events</td>
</tr>
<tr>
<td>errors</td>
<td>errors</td>
</tr>
<tr>
<td>events</td>
<td>mda/rtrid/bfd/vrf</td>
</tr>
<tr>
<td>ha</td>
<td>startup/HA/NSF</td>
</tr>
<tr>
<td>hello</td>
<td>hello events/pkts</td>
</tr>
<tr>
<td>idb</td>
<td>interface</td>
</tr>
<tr>
<td>pkt</td>
<td>I/O packets</td>
</tr>
<tr>
<td>rib</td>
<td>rib batching</td>
</tr>
</tbody>
</table>
### OSPFv3 Commands

#### Name

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>spf</td>
<td>spf/topology</td>
</tr>
<tr>
<td>spf_cycle</td>
<td>spf/topology detail</td>
</tr>
<tr>
<td>te</td>
<td>mpls-te</td>
</tr>
<tr>
<td>test</td>
<td>testing info</td>
</tr>
<tr>
<td>mq</td>
<td>message queue info</td>
</tr>
</tbody>
</table>

#### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

#### Examples

This example shows how to set 1024 error trace entries:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router ospf 3 osp3
RP/0/RP0/CPU0:router(config-ospf3)#trace size errors ?
  0  disable trace
  256 trace entries
  512 trace entries
  1024 trace entries
  2048 trace entries
  4096 trace entries
  8192 trace entries
  16384 trace entries
  32768 trace entries
  65536 trace entries
RP/0/RP0/CPU0:router(config-ospf3)#trace size errors 1024
```
transmit-delay (OSPFv3)

To set the estimated time required to send a link-state update packet on the interface, use the `transmit-delay` command in an appropriate configuration mode. To return to the default value, use the `no` form of this command.

```
transmit-delay seconds
no transmit-delay seconds
```

**Syntax Description**
- `seconds` Time (in seconds) required to send a link-state update. Range is 1 to 65535 seconds.

**Command Default**
1 second

**Command Modes**
- Process configuration
- Area configuration
- Interface configuration
- Virtual-link configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Link-state advertisements (LSAs) in the update packet must have their ages incremented by the amount specified in the `seconds` argument before transmission. The value assigned should take into account the transmission and propagation delays for the interface.

If the delay is not added before transmission over a link, the time in which the LSA propagates over the link is not considered. This setting has more significance on very low-speed links.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure a transmit delay of 3 seconds for tenGigE interface 0/3/0/0:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 1
RP/0/RP0/CPU0:router(config-ospfv3)# area 0
RP/0/RP0/CPU0:router(config-ospfv3-ar)# interface tenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-ospfv3-ar-if)# transmit-delay 3
```
**virtual-link (OSPFv3)**

To define an Open Shortest Path First Version 3 (OSPFv3) virtual link, use the `virtual-link` command in area configuration mode. To remove a virtual link, use the `no` form of this command.

```
virtual-link router-id
no virtual-link
```

**Syntax Description**

`router-id` Router ID associated with the virtual link neighbor. The router ID appears in the `show ospfv3` display. This value must be entered in 32-bit dotted-decimal notation, similar to an IP Version 4 (IPv4) address. There is no default.

**Command Default**

No virtual links are defined.

**Command Modes**

Area configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0      | This command was introduced.

**Usage Guidelines**

In OSPFv3, when there exists a path through another non-backbone area over which the virtual link can function, all areas must be connected to a backbone area. If the connection to the backbone is lost, it can be repaired by establishing a virtual link.

Virtual links, which are defined in the submode of the area they transit, are in effect virtual point-to-point interfaces belonging to area 0 (the backbone). The virtual links inherit parameter values from the backbone area, rather than the transit area in which they are defined.

Each virtual link neighbor must include the router ID of the virtual link neighbor for the link to be properly established. Use the `show ospfv3` command to display the router ID of an OSPFv3 process.

Use the `virtual-link` command to place the router in virtual-link configuration mode (`config-router-ar-vl`), from which you can configure virtual-link-specific settings. Commands configured under this mode (such as the `transmit-delay` command) are automatically bound to that virtual link.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to establish a virtual link with default values for all optional parameters:

```
RP/0/RP0/CPU0:router(config)# router ospfv3 201
RP/0/RP0/CPU0:router(config-ospfv3)# area 1
RP/0/RP0/CPU0:router(config-ospfv3-ar)# virtual-link 10.3.4.5
```
virtual-link (OSPFv3)
RIB Commands

This module describes the commands used to display and clear information in the Routing Information Base (RIB).

For detailed information about RIB concepts, configuration tasks, and examples, see the Implementing RIB on Cisco NCS 5000 Series Routers module in *Routing Configuration Guide for Cisco NCS 5000 Series Routers*.

Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

- address-family next-hop dampening disable, on page 467
- clear route, on page 468
- maximum prefix (RIB), on page 470
- lcc, on page 471
- rcc, on page 472
- recursion-depth-max, on page 473
- router rib, on page 474
- rump always-replicate, on page 475
- show lcc statistics, on page 476
- show rcc, on page 478
- show rcc statistics, on page 480
- show rib, on page 482
- show rib afi-all, on page 484
- show rib attributes, on page 486
- show rib client-id, on page 487
- show rib clients, on page 489
- show rib extcomms, on page 491
- show rib firsthop, on page 493
- show rib history, on page 495
- show rib next-hop, on page 497
- show rib opaquers, on page 499
- show rib protocols, on page 501
- show rib recursion-depth-max, on page 503
• show rib statistics, on page 505
• show rib tables, on page 507
• show rib trace, on page 509
• show rib vpn-attributes, on page 511
• show route, on page 513
• show route backup, on page 520
• show route best-local, on page 523
• show route connected, on page 525
• show route local, on page 527
• show route longer-prefixes, on page 529
• show route next-hop, on page 531
• show route quarantined, on page 533
• show route resolving-next-hop, on page 535
• show route static, on page 537
• show route summary, on page 539
address-family next-hop dampening disable

To disable Routing Information Base (RIB) next-hop dampening, use the `address-family next-hop dampening disable` command in XR Config mode. To enable RIB next-hop dampening, use the `no` form of this command.

`address-family {ipv4|ipv6} next-hop dampening disable`

`no address-family {ipv4|ipv6} next-hop dampening disable`

**Syntax Description**

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>Specifies IP Version 4 (IPv4) address prefixes.</td>
</tr>
<tr>
<td>ipv6</td>
<td>Specifies IP Version 6 (IPv6) address prefixes.</td>
</tr>
</tbody>
</table>

**Command Default**

RIB next-hop dampening is enabled.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to disable RIB next-hop dampening for IPv6 address families:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router rib
RP/0/RP0/CPU0:router(config-rib)# address-family ipv6 next-hop dampening disable
```
clear route

To clear routes from the IP routing table, use the clear route command in XR EXEC mode.

clear route  {ipv4|ipv6|afi-all|safi-all}  {unicast|safi-all}  [topology topo-name]  [ip-address  mask]

Syntax Description

ipv4  Specifies IP Version 4 address prefixes.

ipv6  Specifies IP Version 6 address prefixes.

afi -all  Specifies IP Version 4 and IP Version 6 address prefixes.

safi -all  Specifies unicast address prefixes.

safi-unicast  Specifies unicast and multicast address prefixes.

topology topo-name  (Optional) Specifies topology table information and name of the topology table.

ip-address node-id  (Optional) Clears hardware resource counters from the designated node. The node-id argument is entered in the rack/slot/module notation.

ip-address  Network IP address about which routing information should be displayed.

mask  Network mask specified in either of two ways:

Network mask can be a four-part, dotted-decimal address. For example, 255.0.0.0 indicates that each bit equal to 1 means the corresponding address bit is a network address.

Network mask can be indicated as a slash (/) and number. For example, /8 indicates that the first 8 bits of the mask are 1s, and the corresponding bits of the address are the network address.

Command Default

Local routes received from RIB are displayed for default ipv4 unicast vrf.

Command Modes

XR EXEC mode

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

Use the clear route command to clear routes from an IP routing table to a specific network, a matching subnet address, or all routes.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read, write</td>
</tr>
</tbody>
</table>
RIB Commands

Examples

The following example shows how to remove all routes matching the subnet address 192.168.2.0 and mask 255.255.255.0 from the IPv4 unicast routing table:

RP/0/RP0/CPU0:router# clear route ipv4 unicast 192.168.2.0 255.255.255.0

The following example shows how to remove all routes from the IPv4 unicast routing table:

RP/0/RP0/CPU0:router# clear route ipv4 unicast
maximum prefix (RIB)

To set the prefix limit for the VPN routing and forwarding (VRF) instance, use the `maximum prefix` command in global VRF address family configuration mode. To set the prefix limits to the default values, use the `no` form of this command.

```
maximum prefix maximum [mid-threshold]
no maximum prefix
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>maximum</code></td>
<td>Maximum number of prefixes allowed in the VRF instance. Range is 32 to 2000000.</td>
</tr>
<tr>
<td><code>mid-threshold</code></td>
<td>(Optional) Integer specifying at what percentage of the <code>maximum</code> argument value the software starts to generate a Simple Network Management Protocol (SNMP) trap. Range is 1 to 100.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command Default</th>
<th>No default behavior or values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Modes</td>
<td>Global VRF address family configuration</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command History</th>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

| Usage Guidelines | Use the `maximum prefix` command to configure a maximum number of prefixes that a VRF instance is allowed to receive. |

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

| Examples | The following example shows how to set the maximum number of prefixes allowed to 1000: |

```
RP/0/RP0/CPU0:router(config)# vrf vrf-A
RP/0/RP0/CPU0:router(config-vrf)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-vrf-af)# maximum prefix 1000
```

A maximum number of routes is applicable to dynamic routing protocols as well as static or connected routes. When maximum prefix is configured, an `syslog` message is generated in the following conditions:

1. if the number of routes is above the threshold when “maximum prefix” configuration has been committed
2. if the number routes reaches the configured “maximum prefix” values for the VRF
lcc  

To enable Label Consistency Checker (lcc) background scan for IPv6 or IPv4 labels, use the `lcc enable` command in XR Config mode. To disable lcc background scan, use the `no` for of this command.

```
lcc {ipv4|ipv6} unicast {enable|period milliseconds}
no lcc {ipv4|ipv6} unicast {enable|period milliseconds}
```

**Syntax Description**
- `ipv4` Specifies IP Version 4 address prefixes.
- `ipv6` Specifies IP Version 6 address prefixes.
- `unicast` Specifies unicast address prefixes.
- `period milliseconds` Specifies the period between scans in milliseconds. Range is 100 to 600000 milliseconds.

**Command Default**
Label consistency checker is disabled.

**Command Modes**
XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>read, write</td>
</tr>
<tr>
<td>ipv6</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to enable lcc for IPv6 lables:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#lcc ipv6 unicast enable
```
rcc

To enable Route Consistency Checker (rcc) background scan for IPv6 or IPv4 routes, use the rcc enable command in XR Config mode. To disable rcc background scan, use the no form of this command.

```
rcc {ipv4|ipv6} unicast {enable|period milliseconds}
no rcc {ipv4|ipv6} unicast {enable|period milliseconds}
```

**Syntax Description**
- **ipv4** Specifies IP Version 4 address prefixes.
- **ipv6** Specifies IP Version 6 address prefixes.
- **unicast** Specifies unicast address prefixes.
- **period milliseconds** Specifies the period between scans in milliseconds. Range is 100 to 600000 milliseconds.

**Command Default**
Route consistency checker is disabled.

**Command Modes**
XR Config mode

**Command History**
- **Release 6.0**
  - This command was introduced.

**Usage Guidelines**
Use the period option to control how often the scan be triggered. Each time the scan is triggered, the background scan process resumes verification from where it was left out and sends one buffer’s worth of routes to the forwarding information base (FIB).

**Task ID**
- **Task ID**
  - **ipv4**
    - read, write
  - **ipv6**
    - read, write

This example shows how to configure rcc for IPv6 unicast:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#rcc ipv6 unicast enable
```

This example shows how to enable rcc with a scan period of 500 milliseconds for IPv6 unicast:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#rcc ipv6 unicast period 500
```
recursion-depth-max

To set the maximum depth for route recursion checks, use the `recursion-depth-max` command in XR Config mode. To set the recursion checks to the default value, use the `no` form of this command.

```
recursion-depth-max maximum
no recursion-depth-max maximum
```

**Syntax Description**

- `maximum` Maximum depth for recursion checks. Range is 5 to 16.

**Command Default**

The default recursion depth is 128.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `recursion-depth-max` command to configure a specific maximum number of recursion checks in the range of 5 to 16.

**Examples**

The following example shows how to set the maximum depth for route recursion checks to 12:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router rib
RP/0/RP0/CPU0:router(config-rib)# recursion-depth-max 12
```
To enter Routing Information Base (RIB) configuration mode, use the `router rib` command in XR Config mode. To remove all RIB configurations and terminate the RIB routing process, use the `no` form of this command.

```
router rib
no router rib
```

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

Router configuration mode is not enabled.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>bgp</td>
<td>read, write</td>
</tr>
<tr>
<td>ospf</td>
<td>read, write</td>
</tr>
<tr>
<td>hsrp</td>
<td>read, write</td>
</tr>
<tr>
<td>isis</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to enter RIB configuration mode:

```
RP/0/RP0/CPU0:router(config)# router rib
```
rump always-replicate

To enable replication from uRIB to muRIB as usual even after features such as MTR are configured, use the rump always-replicate command in XR Config mode. To disable replication from uRIB to muRIB, use the no form of this command.

```
rump always-replicate [access-list]
no rump always-replicate [access-list]
```

**Syntax Description**

access-list-name  (Optional) Name of the access list.

**Command Default**

Replication from uRIB to muRIB is enabled.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Configuring the rump always-replicate command allows routers in a network to be upgraded to multitopology routing gradually without a flag day where all routers need to be configured at the same time without major service disruption. When rump always-replicate is configured, replicated routes are added into the muRIB with the lowest admin distance. So if protocols are populating the muRIB, they continue to do so. For the same route, protocol routes win over replicated routes because of higher admin distance.

If an unwanted more specific route comes from the uRIB, optionally provide an access list through which the replicated routes are run. If the route passes the access list, the route is replicated by RUMP.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to enable replication from uRIB to muRIB:

```
RP/0/RP0/CPU0:router(config)# router rib
RP/0/RP0/CPU0:router(config-rib)# address-family ipv4
RP/0/RP0/CPU0:router(config-rib-afi)# rump always-replicate
```
show lcc statistics

To view results of a label consistency checker (lcc) background scan, use the **show lcc statistics** command in XR EXEC mode.

```
show lcc {ipv4|ipv6} unicast statistics {scan-id|summary}
```

**Syntax Description**

- **ipv4** IPv4 address prefix.
- **ipv6** IPv6 address prefix.
- **unicast** Specifies unicast address prefix.
- **scan-id scan-id-value** Specifies the scan ID value. The range is between <0-100000>.
- **summary** Displays a summary of the BG route consistency check statistics.

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

- **ipv4 read**
- **ipv6 read**

This example shows background scan statistics for AFI-SAFI mpls6-unicast:

```
RP/0/RP0/CPU0:router#show lcc ipv6 unicast statistics
```

Background Scan Statistics for AFI-SAFI mpls6-unicast:
```
--------------------------------------------------
Scan enabled: False
Current scan-id: 0
Scan triggered: False
Configured period: 60
Current period: 0
Paused by range scan: False
Paused by route churn: False
Paused by error scan: False
Last data sent: 0 entries
Default route churn: 10
Current route churn: 0
Route churn last calculated at Dec 31 16:00:00.000
Damping percent: 70
```
Logs stored for background scan ids:

Log for AFI-SAFI mplsV6-unicast:
---------------------------------

End Of Logs

This example shows background scan statistics for AFI-SAFI mplsV4-unicast:

RP/0/RP0/CPU0:router#show lcc ipv4 unicast statistics

Background Scan Statistics for AFI-SAFI mplsV4-unicast:
-----------------------------------------------------
Scan enabled: False
Current scan-id: 0 Scan triggered: False
Configured period: 60 Current period: 0
Paused by range scan: False
Paused by route churn: False
Paused by error scan: False

Last data sent: 0 entries Damping percent: 70
Default route churn: 10 Current route churn: 0
Route churn last calculated at Dec 31 16:00:00.000

Logs stored for background scan ids:

Log for AFI-SAFI mplsV4-unicast:
---------------------------------

End Of Logs
**show rcc**

To display route consistency checker (RCC) information, use the `show rcc` command in XR EXEC mode.

```
show rcc {ipv4|ipv6} unicast [prefix netmask ]
```

**Syntax Description**

- `ipv4` Specifies IP Version 4 address prefixes.
- `ipv6` Specifies IP Version 6 address prefixes.
- `unicast` Specifies unicast address prefixes.
- `prefix` (Optional) Starting prefix.
- `netmask` (Optional) Network mask.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Examples**

The following is sample output from the `show rcc` command:

```
RP/0/RP0/CPU0:router# show rcc ipv4 unicast statistics
Thu Mar 26 13:47:28.391 IST
Background Scan Summary
------------------------
Scan enabled: False Last scan-id: 0
Configured period: 15000 Current period: 0
Paused By:
route churn:False on-demand scan:False error scan:False
Last data sent: 0 entries Damping percent: 69
Default route churn: 100 Current route churn: 0
Route churn last calculated at Never
Logs last cleared at Never
Scan paused by ISSU False
```
Logs stored for background scan ids:

Scan Logs
---------
Legend:
  ? - Currently Inactive Node, ! - Non-standard SVD Role
  * - Node did not reply

End of Logs
show rcc statistics

To view results of a route consistency checker (rcc) background scan, use the `show rcc statistics` command in XR EXEC mode.

```
show rcc {ipv4|ipv6} unicast statistics {scan-id|summary}
```

**Syntax Description**

- `ipv4` IPv4 address prefix.
- `ipv6` IPv6 address prefix.
- `unicast` Specifies unicast address prefixes.
- `scan-id scan-id-value` Specifies the scan ID value. The range is between <0-100000>.
- `summary` Displays a summary of the BG route consistency check statistics.

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

- **Release**
  - Release 6.0
  - This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>read</td>
</tr>
<tr>
<td>ipv6</td>
<td>read</td>
</tr>
</tbody>
</table>

This example shows background scan statistics for AFI-SAFI IPv6 unicast:

```
RP/0/RP0/CPU0:router# show rcc ipv6 unicast statistics
Background Scan Statistics for AFI-SAFI ipv6-unicast:
-----------------------------------------------
Scan enabled: False
Current scan-id: 0 Scan triggered: False
Configured period: 60 Current period: 0
Paused by range scan: False
Paused by route churn: False
Paused by error scan: False

Last data sent: 0 entries Damping percent: 70
Default route churn: 10 Current route churn: 0
Route churn last calculated at Dec 31 16:00:00.000
```
Logs stored for background scan ids:

Log for AFI-SAFI ipv6-unicast:
================================
End Of Logs

This example shows background scan statistics for AFI-SAFI IPv4 unicast:

RP/0/RP0/CPU0:router#show rcc ipv4 unicast statistics

Background Scan Statistics for AFI-SAFI ipv4-unicast:
======================================================
Scan enabled: False
Current scan-id: 0  Scan triggered: False
Configured period: 60  Current period: 0
Paused by range scan: False
Paused by route churn: False
Paused by error scan: False

Last data sent: 0 entries  Damping percent: 70
Default route churn: 10  Current route churn: 0
Route churn last calculated at Dec 31 16:00:00.000

Logs stored for background scan ids:

Log for AFI-SAFI ipv4-unicast:
================================
End Of Logs
**show rib**

To display Routing Information Base (RIB) data, use the `show rib` command in XR EXEC mode.

```
show rib {ipv4|ipv6} {unicast}[{firsthop[ {type interface-path-id}]|next-hop[ {type interface-path-id}]}|opaques |{attribute |ip-nexthop|summary}|protocols[ {standby}]}|statistics[ {name}|{standby}]|topology[ {topo-name|all} ]}
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies IP Version 4 address prefixes.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td>unicast</td>
<td>(Optional) Specifies unicast address prefixes. This is the default.</td>
</tr>
<tr>
<td>firsthop</td>
<td>(Optional) Specifies registered first-hop notification addresses.</td>
</tr>
<tr>
<td>type</td>
<td>Interface type. For more information, use the question mark (?) online help function.</td>
</tr>
<tr>
<td>interface-path-id</td>
<td>Identifies a physical interface or a virtual interface.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>Use the show interfaces command to see a list of all possible interfaces currently configured on the router.</td>
</tr>
<tr>
<td></td>
<td>For more information about the syntax for the router, use the question mark (?) online help function.</td>
</tr>
<tr>
<td>next-hop</td>
<td>(Optional) Specifies registered next-hop notification addresses.</td>
</tr>
<tr>
<td>opaques</td>
<td>(Optional) Specifies opaque data installed in the RIB.</td>
</tr>
<tr>
<td>attribute</td>
<td>(Optional) Specifies opaque attributes installed in the RIB.</td>
</tr>
<tr>
<td>ip-nexthop</td>
<td>(Optional) Specifies P next-hop data installed in the RIB.</td>
</tr>
<tr>
<td>summary</td>
<td>(Optional) Specifies a summary of opaque data installed in the RIB.</td>
</tr>
<tr>
<td>protocols</td>
<td>(Optional) Specifies registered protocols.</td>
</tr>
<tr>
<td>statistics name</td>
<td>(Optional) Specifies RIB statistics of a given name.</td>
</tr>
<tr>
<td>standby</td>
<td>(Optional) Specifies standby information.</td>
</tr>
<tr>
<td>topology topo-name</td>
<td>(Optional) Specifies topology table information and name of the topology table.</td>
</tr>
<tr>
<td>all</td>
<td>(Optional) Specifies that all topology table information should be displayed.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode
Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>read</td>
</tr>
</tbody>
</table>

Examples

The following example illustrates the `show rib` command:

```
RP/0/RSP0RP0/CP00:router# show rib

ipv4 multicast

topology
BLUE

RP/0/RSP0RP0/CP00:router# show rib topology BLUE ipv4 multicast protocols
Protocol  Handle  Instance
isis       0        mt
```
show rib afi-all

To display Routing Information Base (RIB) data for both IPv4 and IPv6 address families, use the `show rib afi-all` command in XR EXEC mode.

```
show rib afi-all [attributes] [client-id] [clients] [extcomms] [firsthop] [history] [next-hop] [opaques] [protocols] [recursion-depth-max] [safi-all] [statistics] [tables] [trace] [unicast]
```

**Syntax Description**

- **attributes** (Optional) Displays all BGP attributes installed in RIB.
- **client-id** (Optional) Displays RIB client ID for longer history of redistributed routes sent to the client.
- **clients** (Optional) Displays RIB clients.
- **extcomms** (Optional) Displays all extended communities installed in RIB.
- **firsthop** (Optional) Displays registered first-hop notification addresses.
- **history** (Optional) Displays redistributed routes sent to RIB clients.
- **next-hop** (Optional) Displays registered next-hop notification addresses.
- **opaques** (Optional) Displays opaque data installed in RIB.
- **protocols** (Optional) Displays registered protocols.
- **recursion-depth-max** (Optional) Displays maximum recursion depth in RIB.
- **safi-all** (Optional) Displays unicast and multicast commands.
- **statistics** (Optional) Displays RIB statistics.
- **tables** (Optional) Displays a list of tables known to RIB.
- **trace** (Optional) Displays RIB trace entries.
- **unicast** (Optional) Displays unicast commands.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.
Examples

The following example illustrates the `show rib afi-all attributes` command:

```plaintext
RP/0/RP0/CPU0:router# show rib afi-all attributes
BGP attribute data in IPv4 RIB:
0 Attributes, for a total of 0 bytes.
BGP attribute data in IPv6 RIB:
0 Attributes, for a total of 0 bytes.
```
show rib attributes

To display Border Gateway Protocol (BGP) attributes installed in the Routing Information Base (RIB), use the \texttt{show rib attributes} command in XR EXEC mode.

\texttt{show rib attributes [summary] [standby]}

**Syntax Description**
- \texttt{summary} (Optional) Displays a summary of BGP attribute data installed in the RIB.
- \texttt{standby} (Optional) Displays standby information.

**Command Default**
No default behavior or values

**Command Modes**
XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**
The following is sample output from the \texttt{show rib attributes} command:

\begin{verbatim}
RP/0/RP0/CPU0:router# show rib attributes
BGP attribute data in IPv4 RIB:
Attribute ID (0x2):size (68)
Attribute ID (0x3):size (52)
Attribute ID (0x4):size (68)
Attribute ID (0x5):size (52)

4 Attributes, for a total of 240 bytes.
Attribute ID : ID assigned for the attribute by BGP
size : size of the attribute data.
\end{verbatim}
show rib client-id

To display Routing Information Base (RIB) redistribution histories, use the `show rib client-id` command in XR EXEC mode.

```
show rib client-id id redistribution history [standby]
```

**Syntax Description**

- `id` ID of the client. Range is 0 to 4294967295.
- `redistribution history` Displays longer history of redistributed routes sent to RIB clients.
- `standby` (Optional) Displays standby information.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rib client-id` command to display a history of the route additions, deletions, and updates sent from RIB to the client.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show rib client-id` command:

```
RP/0/RP0/CPU0:router# show rib client-id 13 redistribution history

PID    JID  Client      Location
Table ID: 0xe0000000
151630  113  bcdl_agent node0_5_CPU0

S 80.80.80.0/24[1/0] update, 5 path(s), 0x0 Jan 31 09:54:57.224
S 80.80.80.0/24[1/0] update, 6 path(s), 0x0 Jan 31 09:53:39.736
S 140.140.140.0/24[1/0] update, 1 path(s), 0x0 Jan 31 09:53:39.729
S 80.80.80.0/24[1/0] update, 5 path(s), 0x0 Jan 30 22:08:38.551
S 140.140.140.0/24 deleted, Jan 30 22:08:38.543
S 80.80.80.0/24[1/0] update, 6 path(s), 0x0 Jan 30 22:03:05.889
S 100.100.100.0/24[1/0] update, 1 path(s), 0x0 Jan 30 22:03:05.880
```
This table describes the significant fields shown in the display.

**Table 85: show rib client-id Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID</td>
<td>Process ID of the client.</td>
</tr>
<tr>
<td>JID</td>
<td>Job ID of the client.</td>
</tr>
<tr>
<td>Client</td>
<td>Client name.</td>
</tr>
<tr>
<td>Location</td>
<td>Location node on which the client is present.</td>
</tr>
</tbody>
</table>
**show rib clients**

To display Routing Information Base (RIB) clients, use the `show rib clients` command in XR EXEC mode.

```
show rib [{afi-all|ipv4|ipv6}] clients [{protocols|redistribution {history}]} [standby]
```

**Syntax Description**

- `afi-all` *(Optional)* Specifies all address families.
- `ipv4` *(Optional)* Specifies IP Version 4 address prefixes. This is the default.
- `ipv6` *(Optional)* Specifies IP Version 6 address prefixes.
- `protocols` *(Optional)* Specifies client protocols.
- `redistribution` *(Optional)* Specifies protocols redistributed by clients.
- `history` *(Optional)* Specifies redistributing routes sent to RIB clients.
- `standby` *(Optional)* Displays standby information.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rib clients` command to display the list of clients who have registered with RIB, what protocol routes they are redistributing, and a history of the routes sent to the client.

The maximum number of redistribution entries is 5000 for Bulk Content Downloader (BCDL) and 500 for other protocols.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show rib clients` command:

```
RP/0/R0/CP0:router# show rib clients
Process Location Client ID Redist Proto
isis node0_5_CP0_0 0 insync insync
ospf node0_5_CP0_0 2 insync insync

RP/0/R0/CP0:router# show rib clients redistribution
isis node0_5_CP0_0
```
This table describes the significant fields shown in the display.

**Table 86: show rib clients Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>Client process name.</td>
</tr>
<tr>
<td>Location</td>
<td>Location where the client process is running.</td>
</tr>
<tr>
<td>Client ID</td>
<td>ID assigned to the client by RIB.</td>
</tr>
<tr>
<td>Redist</td>
<td>Whether the client is redistributing any protocols or not and whether it has read all routes from RIB or not.</td>
</tr>
<tr>
<td>Proto</td>
<td>Whether the protocol has sent all its routes to RIB and signaled update complete or not.</td>
</tr>
</tbody>
</table>

- **Redist**
  - insync—read
  - outsync—not read.

- **Proto**
  - insync—read
  - outsync—not read.
show rib extcomms

To display all extended communities installed in the Routing Information Base (RIB), use the `show rib extcomms` command in XR EXEC mode.

```
show rib [{afi-all|ipv4|ipv6}] extcomms [summary] [standby]
```

**Syntax Description**

- `afi-all` (Optional) Specifies all address families.
- `ipv4` (Optional) Specifies IP Version 4 address prefixes. This is the default.
- `ipv6` (Optional) Specifies IP Version 6 address prefixes.
- `summary` (Optional) Specifies a summary of all extended communities in the RIB.
- `standby` (Optional) Displays standby information.

**Command Default**

No default behavior or values

**Command Modes**

System Admin EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show rib extcomms` command:

```
RP/0/RP0/CPU0:router# show rib extcomms

Extended community data in RIB:

Extended community       Ref count
COST:128:128:41984       1
```
This table describes the significant fields shown in the display.

Table 87: show rib extcomms Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extended Community</td>
<td>Type of extended communities. Different protocols can add different extended communities.</td>
</tr>
<tr>
<td>Ref Count</td>
<td>Number of routes referring to the Extended community.</td>
</tr>
</tbody>
</table>
show rib firsthop

To display registered first-hop notification addresses, use the `show rib firsthop` command in System Admin EXEC mode.

```
show rib [afi-all|ipv4|ipv6] [unicast|safi-all] firsthop [client-name] [type interface-path-id|ip-address / prefix-length|ip-address mask|resolved|unresolved|damped] [summary] [standby]
```

**Syntax Description**

- `afi-all` (Optional) Specifies all address families.
- `ipv4` (Optional) Specifies IP Version 4 address prefixes. This is the default.
- `ipv6` (Optional) Specifies IP Version 6 address prefixes.
- `unicast` (Optional) Specifies unicast address prefixes. This is the default.
- `safi-all` (Optional) Specifies unicast and multicast address prefixes.
- `client-name` (Optional) Name of the RIB client.
- `type` Interface type. For more information, use the question mark (?) online help function.
  - `interface-path-id` Physical interface or virtual interface.

**Note** Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

- `ip-address` (Optional) Network that BGP advertises.
- `/ prefix-length` (Optional) Length of the IP address prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash (/) must precede the decimal value.
- `ip-address mask` (Optional) Network mask applied to the `ip-address` argument.
- `resolved` (Optional) Specifies resolved next-hops.
- `unresolved` (Optional) Specifies unresolved next-hops.
- `damped` (Optional) Specifies next-hops that are damped.
- `summary` (Optional) Specifies a summary of the next-hop information.
- `standby` (Optional) Displays standby information.

**Command Default**

Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command Modes**

System Admin EXEC mode
**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rib firsthop` command to display the list of first hops registered by various clients with RIB and the address and interface through which they are resolved.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show rib firsthop` command:

```
RP/0/RP0/CPU0:router# show rib firsthop

Registered firsthop notifications:
0.0.0.0/0 via 1.1.0.1 - MgmtEth0/5/CPU0/0, ospf/node0_5_CPU0
1.1.0.1/32 via 1.1.0.1 - MgmtEth0/5/CPU0/0, ipv4_static/node0_5_CPU0
1.1.1.1/32 via 1.1.1.1 - MgmtEth0/5/CPU0/0, ipv4_static/node0_5_CPU0
10.10.10.1/32 via 10.10.10.1 - Loopback0, ipv4_static/node0_5_CPU0
15.15.15.1/32 via 10.10.10.1 - Loopback0, ipv4_static/node0_5_CPU0
20.20.20.1/32 via 1.1.1.1 - MgmtEth0/5/CPU0/0, ipv4_static/node0_5_CPU0
30.30.30.1/32 via 1.1.1.2 - MgmtEth0/5/CPU0/0, ipv4_static/node0_5_CPU0
```
show rib history

To display history information for Routing Information Base (RIB) clients, use the `show rib history` command in XR EXEC mode.

**Syntax Description**

```
show rib [{afi-all|ipv4|ipv6}] history [client-id client-id] [standby]
```

- `afi-all` (Optional) Specifies all address families.
- `ipv4` (Optional) Specifies IP Version 4 address prefixes. This is the default.
- `ipv6` (Optional) Specifies IP Version 6 address prefixes.

- `client-id client-id` (Optional) Specifies the ID of the client. Range for `client-id` argument is 0 to 4294967295.

- `standby` (Optional) Displays standby information.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rib history` command to display the list of routes that RIB has sent to various clients.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show rib history` command:

```
RP/0/RP0/CP00:router# show rib history

JID     Client Location           Table ID: 0xe0000000
229      isis node0_5_CPU0
          S 80.80.80.0/24[1/0] update, 6 path(s), 04:32:09
          S 100.100.100.0/24[1/0] update, 1 path(s), 04:32:09
          S 40.40.40.0/24[1/0] update, 1 path(s), 04:32:09
          S 15.15.15.0/24[1/0] update, 1 path(s), 04:32:09

JID     Client Location           Table ID: 0xe0000000
260      ospf node0_5_CPU0
          S 80.80.80.0/24[1/0] update, 6 path(s), 04:32:09
          S 100.100.100.0/24[1/0] update, 1 path(s), 04:32:09
          S 40.40.40.0/24[1/0] update, 1 path(s), 04:32:09
          S 15.15.15.0/24[1/0] update, 1 path(s), 04:32:09
```
This table describes the significant fields shown in the display.

Table 88: show rib history Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>JID</td>
<td>Job ID of the client process.</td>
</tr>
<tr>
<td>Client</td>
<td>Name of the client process.</td>
</tr>
<tr>
<td>Location</td>
<td>Information about where the client process is running.</td>
</tr>
</tbody>
</table>
show rib next-hop

To display registered next-hop notification addresses, use the `show rib next-hop` command in XR EXEC mode.

```
show rib [vrf {vrf-name|all}] [{afi-all|ipv4|ipv6}] [{unicast|multicast|safi-all}] next-hop [client-name] [{type interface-path-id|ip-address /prefix-length|ip-address mask|resolved|unresolved|damped}] [summary] [standby]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vrf {vrf-name</td>
<td>all}</td>
</tr>
<tr>
<td>afi-all</td>
<td>(Optional) Specifies all address families.</td>
</tr>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies IP Version 4 address prefixes. This is the default.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td>unicast</td>
<td>(Optional) Specifies unicast address prefixes. This is the default.</td>
</tr>
<tr>
<td>multicast</td>
<td>(Optional) Specifies multicast address prefixes.</td>
</tr>
<tr>
<td>safi-all</td>
<td>(Optional) Specifies unicast and multicast address prefixes.</td>
</tr>
<tr>
<td>client-name</td>
<td>(Optional) Name of the RIB client.</td>
</tr>
<tr>
<td>type</td>
<td>Interface type. For more information, use the question mark ( ? ) online help function.</td>
</tr>
<tr>
<td>interface-path-id</td>
<td>Physical interface or virtual interface.</td>
</tr>
<tr>
<td>ip-address</td>
<td>(Optional) Network IP address about which routing information should be displayed.</td>
</tr>
<tr>
<td>mask</td>
<td>(Optional) Network mask specified in either of two ways:</td>
</tr>
<tr>
<td>/prefix-length</td>
<td>(Optional) Length of the IP address prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash (/) must precede the decimal value.</td>
</tr>
<tr>
<td>resolved</td>
<td>(Optional) Specifies resolved next-hops.</td>
</tr>
<tr>
<td>Command Default</td>
<td>No default behavior or values</td>
</tr>
<tr>
<td>----------------------</td>
<td>------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Command Modes</td>
<td>XR EXEC mode</td>
</tr>
<tr>
<td>Command History</td>
<td></td>
</tr>
<tr>
<td>Release</td>
<td>Modification</td>
</tr>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
<tr>
<td>Usage Guidelines</td>
<td>Use the show rib next-hop command to display the list of next-hops registered by various clients with the RIB and the address and interface through which they are resolved.</td>
</tr>
<tr>
<td>Task ID</td>
<td></td>
</tr>
<tr>
<td>rib</td>
<td>Operations</td>
</tr>
<tr>
<td>read</td>
<td></td>
</tr>
<tr>
<td>Examples</td>
<td>The following is sample output from the show rib next-hop command:</td>
</tr>
<tr>
<td></td>
<td>RP/0/RP0/CPU0:router# show rib next-hop</td>
</tr>
<tr>
<td></td>
<td>Registered nexthop notifications:</td>
</tr>
<tr>
<td></td>
<td>0.0.0.0/0 via 172.29.52.1 - MgmtEth0/RP1/CPU0/0, ospf/node0_RP0_CPU0</td>
</tr>
<tr>
<td></td>
<td>172.29.52.1/32 via 172.29.52.1 - MgmtEth0/RP1/CPU0/0, ipv4_static/node0_RP0_CPU0</td>
</tr>
</tbody>
</table>
show rib opaques

To display opaque data installed in the Routing Information Base (RIB), use the `show rib opaques` command in XR EXEC mode.

```
show rib [afi-all|ipv4|ipv6] [unicast|safi-all] opaques [attribute|ip-nexthop|summary|tunnel-nexthop] [rib-client-name] [standby]
```

**Syntax Description**

- **afi-all** (Optional) Specifies all address families.
- **ipv4** (Optional) Specifies IP Version 4 address prefixes. This is the default.
- **ipv6** (Optional) Specifies IP Version 6 address prefixes.
- **unicast** (Optional) Specifies unicast address prefixes. This is the default.
- **safi-all** (Optional) Specifies unicast and multicast address prefixes.
- **attribute** Displays opaque attributes installed in the RIB.
- **ip-nexthop** Displays IP next-hop data installed in the RIB.
- **summary** Displays a summary of opaque data installed in the RIB.
- **tunnel-nexthop** Displays tunnel next-hop opaque data installed in the RIB.
- **rib-client-name** (Optional) Name of the RIB client.
- **standby** (Optional) Displays standby information.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If information is not used by the RIB server process, it is viewed as opaque data. Use the `show rib opaques` command to display opaque data installed in the RIB.

**Examples**

The following is sample output from the `show rib opaques` command:

```
RP/0/RP0/CPU0:router# show rib opaques safi-tunnel
```
Summary of safi tunnel opaque data in IPv4 RIB:

Opaque key: 1:10.1.0.2
Opaque data:
Tunnel Encap - ifhandle=0x1000180, type=L2TPv3, Params={Session-id=0x1EB1127C, `CookieLen=8, Cookie=0xA73A3E0AFCD419A6} Opaque key: 65535:10.0.101.1 Opaque data:

RP/0/RP0/CPU0:router# show rib ipv6 opaques tunnel-nexthop

Summary of 6PE/6VPE IP over tunnel nexthop opaque data in IPv6 RIB:

Opaque key: 1::ffff:10.1.0.2
Opaque key: 65535::ffff:10.0.101.1
Opaque key: 65535::ffff:10.0.101.2
Opaque key: 65535::ffff:10.0.101.3
Opaque key: 65535::ffff:10.0.101.4
Opaque key: 65535::ffff:10.0.101.5

Note

safi-tunnel keyword is not supported.

This table describes the significant fields shown in the display.

Table 89: show rib opaques Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opaque key</td>
<td>Unique key for the opaque data as populated by the protocol client.</td>
</tr>
<tr>
<td>Opaque data</td>
<td>Data for the given key.</td>
</tr>
</tbody>
</table>
show rib protocols

To display protocols registered for route addition, use the `show rib protocols` command in XR EXEC mode.

```
show rib [vrf {vrf-name|all}] [{afi-all|ipv4|ipv6}] [{unicast|multicast|safi-all}] protocols [standby]
```

**Syntax Description**
- `vrf {vrf-name | all}`: (Optional) Specifies a particular VPN routing and forwarding (VRF) instance or all VRF instances.
- `afi-all`: (Optional) Specifies all address families.
- `ipv4`: (Optional) Specifies IP Version 4 address prefixes. This is the default.
- `ipv6`: (Optional) Specifies IP Version 6 address prefixes.
- `unicast`: (Optional) Specifies unicast address prefixes. This is the default.
- `multicast`: (Optional) Specifies multicast address prefixes.
- `safi-all`: (Optional) Specifies unicast and multicast address prefixes.
- `standby`: (Optional) Displays standby information.

**Command Default**
Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command History**
```
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>
```

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show rib protocols` command:

```
RP/0/RP0/CPU0:router# show rib protocols

Protocol   Handle Instance
-------------
isis        0     rib
connected   1
static      2
local       3
bgp         4     102
ospf        5     1
```
This table describes the significant fields shown in the display.

*Table 90: show rib protocols Field Descriptions*

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protocol</td>
<td>Name of the protocol.</td>
</tr>
<tr>
<td>Handle</td>
<td>Handle assigned to the protocol instance.</td>
</tr>
<tr>
<td>Instance</td>
<td>Protocol instance.</td>
</tr>
</tbody>
</table>
show rib recursion-depth-max

To display the maximum recursion depth in the Routing Information Base (RIB), use the `show rib recursion-depth-max` command in XR EXEC mode.

```
show rib [{afi-all|ipv4|ipv6}] recursion-depth-max [standby]
```

**Syntax Description**

- `afi-all` (Optional) Specifies all address families.
- `ipv4` (Optional) Specifies IP Version 4 address prefixes. This is the default.
- `ipv6` (Optional) Specifies IP Version 6 address prefixes.
- `standby` (Optional) Displays standby information.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

- **Release 6.0**: This command was introduced.

**Usage Guidelines**

Use the `show rib recursion-depth-max` command to display the maximum recursion depth for RIB. Recursion depth is the number of next-hops that can be specified.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show rib recursion-depth-max` command:

```
RP/0/RP0/CPU0:router# show rib recursion-depth-max

IPv4:
-----
Maximum recursion depth in RIB:

   Configured: 12
   In Use: 128

IPv6:
-----
Maximum recursion depth in RIB:

   Configured: 12
   In Use: 128
```
This table describes the significant fields shown in the display.

**Table 91: show rib recursion-depth-max Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configured</td>
<td>Value of maximum recursion depth currently configured.</td>
</tr>
<tr>
<td>In Use</td>
<td>Value of maximum recursion depth RIB is using. This value can be different from the configured value because RIB has to be restarted after the configuration is changed for the new configuration to be effective.</td>
</tr>
</tbody>
</table>
**show rib statistics**

To display Routing Information Base (RIB) statistics, use the `show rib statistics` command in XR EXEC mode.

```
show rib [afi-all|ipv4|ipv6] [unicast|safi-all] statistics [client-name] [standby]
```

**Syntax Description**

- **afi-all** (Optional) Specifies all address families.
- **ipv4** (Optional) Specifies IP Version 4 address prefixes. This is the default.
- **ipv6** (Optional) Specifies IP Version 6 address prefixes.
- **unicast** (Optional) Specifies unicast address prefixes. This is the default.
- **safi-all** (Optional) Specifies unicast and multicast address prefixes.
- **client-name** (Optional) Name of the RIB client.
- **standby** (Optional) Displays standby information.

**Command Default**

Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rib statistics` command to display RIB statistics. The statistics include requests sent from the clients to the RIB and the information redistributed to the client.

RIB maintains counters for all requests sent from a client including:

- Route operations
- Table registrations
- Next-hop registrations
- Redistribution registrations
- Attribute registrations
- Synchronization completion

RIB also maintains the results of the requests.

**Examples**

The following is sample output from the `show rib statistics` command:

```
Routing Command Reference for Cisco NCS 5000 Series Routers
505
```
RP/0/RP0/CPU0:router# show rib statistics

RIB Statistics:
Received 142 batch messages
  137 route operations, 0 attribute operations
  0 opaque operations
  11 complete operations, 0 convergent operations
Results of the batch message received:
  142 successes
  0 forward references, 0 invalid client id, 0 unknown errors
  0 memory allocation errors, 0 client lookup errors, table lookup errors 0
  0 proto lookup errors, 0 client proto lookup errors
ipv4_connected/node0_RP0_CPU0 last performed route operation
  with status BATCH_SUCCESS at Jun 26 21:43:33.601

Received 217422 light weight messages
  4 route add requests, 2 route delete requests
  10 protocol registered, 1 protocol unregistered
  0 protocol modify, 0 protocol purged
  14 protocol redistributions, 0 unregistered protocol redistributions
  0 reset protocol redistributions
  3 first hop registered, 1 first hop unregistered
  3 advertisements, 0 unregistered advertisement
  57 bind data, 97 update completes, 217230 other requests
udp/node0_RP0_CPU0 last performed firsthop lookup operation
  with status success at Jun 27 10:09:59.990

Received 0 nexthop batch messages
  0 successes
  0 inits
  0 registers, 0 unregisters
  0 register complete, 0 sync unregistered, 0 batch finished

This table describes the significant fields shown in the display.

Table 92: show rib statistics Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Received</td>
<td>Statistics received including batch messages and route, attribute, complete, and convergent operations.</td>
</tr>
<tr>
<td>Results of the batch message received</td>
<td>Batch message results.</td>
</tr>
<tr>
<td>Received ( n ) light weight messages</td>
<td>Number of lightweight API messages sent from RIB clients.</td>
</tr>
<tr>
<td>Received ( n ) nexthop batch messages</td>
<td>Number of batch API messages sent from RIB clients received by the RIB.</td>
</tr>
</tbody>
</table>
show rib tables

To display all tables known to the Routing Information Base (RIB), use the **show rib tables** command in XR EXEC mode.

```
show rib [{afi-all|ipv4|ipv6}] tables [summary] [standby]
```

**Syntax Description**

- `afi-all` (Optional) Specifies all address families.
- `ipv4` (Optional) Specifies IP Version 4 address prefixes. This is the default.
- `ipv6` (Optional) Specifies IP Version 6 address prefixes.
- `summary` (Optional) Displays summary table information.
- `standby` (Optional) Displays standby information.

**Command Default**
No default behavior or values

**Command Modes**
XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the **show rib tables** command to display all tables known to the RIB, including table attributes. Attributes include address family, and maximum prefix information.

**Task ID**
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the **show rib tables** command when entered without an address:

```
RP/0/RP0/CPU0:router# show rib tables

Codes: N - Prefix Limit Notified, F - Forward Referenced
       D - Table Deleted, C - Table Reached Convergence

<table>
<thead>
<tr>
<th>VRF</th>
<th>SAFI</th>
<th>Table ID</th>
<th>PrfxLmt</th>
<th>PrfxCnt</th>
<th>TblVersion</th>
<th>N</th>
<th>F</th>
<th>D</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>default</td>
<td>uni</td>
<td>0xe0000000</td>
<td>2000000</td>
<td>72</td>
<td>137</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
<tr>
<td>default</td>
<td>multi</td>
<td>0xe0100000</td>
<td>2000000</td>
<td>0</td>
<td>0</td>
<td>N</td>
<td>N</td>
<td>N</td>
<td>N</td>
</tr>
</tbody>
</table>
```
This table describes the significant fields shown in the display.

**Table 93: show rib tables Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SAFI</td>
<td>Subaddress family instance.</td>
</tr>
<tr>
<td>Table ID</td>
<td>ID of the RIB table.</td>
</tr>
<tr>
<td>PrfxLmt</td>
<td>Configured prefix limit for the RIB table.</td>
</tr>
<tr>
<td>PrfxCnt</td>
<td>Number of configured prefixes in the RIB table.</td>
</tr>
<tr>
<td>TblVersion</td>
<td>Tables version number.</td>
</tr>
<tr>
<td>N</td>
<td>Message sent when prefix limit is exceeded.</td>
</tr>
<tr>
<td>F</td>
<td>Forward referenced. If Y is indicated, a table has been created by RIB because a client has registered for the table, but RIB has not heard from the router space infrastructure (RSI) about the table. RSI manages the tables.</td>
</tr>
<tr>
<td>D</td>
<td>If Y is indicated, the table has been deleted in the RSI but RIB has not cleared the information.</td>
</tr>
<tr>
<td>C</td>
<td>Table reached convergence.</td>
</tr>
</tbody>
</table>
# show rib trace

To display all Routing Information Base (RIB) library call tracer (ltrace) entries, use the `show rib trace` command in XR EXEC mode.

```
show rib [[afi-all]ipv4]ipv6] trace [[clear|counts|event-manager|startup|sync|timing] ] [[unique|wrapping] ] [last entries] [hexdump] [reverse] [tailif] [stats] [verbose] [{file name original location node-id]location {all/node-id}}]
```

## Syntax Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>afi-all</code></td>
<td>(Optional) Specifies all address families.</td>
</tr>
<tr>
<td><code>ipv4</code></td>
<td>(Optional) Specifies IP Version 4 address prefixes. This is the default.</td>
</tr>
<tr>
<td><code>ipv6</code></td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td><code>counts clear</code></td>
<td>(Optional) Displays route clear trace entries.</td>
</tr>
<tr>
<td><code>counts</code></td>
<td>(Optional) Displays counts trace entries.</td>
</tr>
<tr>
<td><code>event-manager</code></td>
<td>(Optional) Displays RIB event manager trace entries.</td>
</tr>
<tr>
<td><code>startup</code></td>
<td>(Optional) Displays RIB startup trace entries.</td>
</tr>
<tr>
<td><code>sync</code></td>
<td>(Optional) Displays client synchronization trace entries.</td>
</tr>
<tr>
<td><code>timing</code></td>
<td>(Optional) Displays timing trace entries.</td>
</tr>
<tr>
<td><code>unique</code></td>
<td>(Optional) Displays unique entries with counts.</td>
</tr>
<tr>
<td><code>wrapping</code></td>
<td>(Optional) Displays wrapping entries.</td>
</tr>
<tr>
<td><code>last entries</code></td>
<td>(Optional) Displays a specified number of the last entries. Range is 1 to 4294967295.</td>
</tr>
<tr>
<td><code>hexdump</code></td>
<td>(Optional) Displays traces in hexadecimal format.</td>
</tr>
<tr>
<td><code>reverse</code></td>
<td>(Optional) Displays the latest traces first.</td>
</tr>
<tr>
<td><code>tailif</code></td>
<td>(Optional) Displays new traces as they are added.</td>
</tr>
<tr>
<td><code>stats</code></td>
<td>(Optional) Displays statistics.</td>
</tr>
<tr>
<td><code>verbose</code></td>
<td>(Optional) Displays internal debugging information.</td>
</tr>
<tr>
<td><code>file name original location node-id</code></td>
<td>(Optional) Displays trace entries for a specific file for the designated node. The <code>node-id</code> argument is entered in the <code>rack/slot/module</code> notation.</td>
</tr>
<tr>
<td><code>location</code></td>
<td>{ all</td>
</tr>
</tbody>
</table>

## Command Default

No default behavior or values
**show rib trace**

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the *show rib trace* command:

```
RP/0/RP0/CPU0:router# show rib trace

1784 wrapping entries (13312 possible, 0 filtered, 1784 total)
Mar 16 14:59:27.947 rib/ipv4_rib/rib-startup 0/RSP0RP0/CPU0 t1 Create: Management thread
Mar 16 14:59:27.959 rib/ipv4_rib/rib-startup 0/RSP0RP0/CPU0 t2 Create: Management event manager
Mar 16 14:59:28.346 rib/ipv4_rib/rib-io 0/RSP0RP0/CPU0 t1 Initialise: RIB server
Mar 16 14:59:28.346 rib/ipv4_rib/rib-io 0/RSP0RP0/CPU0 t1 Initialise: Client collection
Mar 16 14:59:28.676 rib/ipv4_rib/rib-io 0/RSP0RP0/CPU0 t1 Initialise: DB collection
Mar 16 14:59:28.693 rib/ipv4_rib/rib-io 0/RSP0RP0/CPU0 t1 Initialise: Timer tree
Mar 16 14:59:28.694 rib/ipv4_rib/rib-io 0/RSP0RP0/CPU0 t1 RUMP: Bind to sysdb /ipc/gl/ipv4-rib/ for protocol notification
Mar 16 14:59:29.102 rib/ipv4_rib/rib-startup 0/RSP0RP0/CPU0 t2 Initialise: Debugging routine
Mar 16 14:59:29.128 rib/ipv4_rib/rib-io 0/RSP0RP0/CPU0 t1 Register: read, select cb functions
Mar 16 14:59:29.137 rib/ipv4_rib/rib-startup 0/RSP0RP0/CPU0 t1 Register: cerrno DLL name lib_rib_error.dll
```

Routing Command Reference for Cisco NCS 5000 Series Routers


**show rib vpn-attributes**

To display all VPN attributes installed in the Routing Information Base (RIB), use the `show rib vpn-attributes` command in XR EXEC mode.

```
show rib [afi-all|ipv4|ipv6] vpn-attributes [summary] [standby]
```

**Syntax Description**

- `afi-all` (Optional) Specifies all address families.
- `ipv4` (Optional) Specifies IP Version 4 address prefixes.
- `ipv6` (Optional) Specifies IP Version 6 address prefixes.
- `summary` (Optional) Displays VPN attribute information.
- `standby` (Optional) Displays standby information.

**Command Default**

The default is IPv4 address prefixes.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

```
Task ID  Operations
rib      read
```

**Examples**

The following is sample output from the `show rib vpn-attributes` command:

```
RP/0/RP0/CPU0:router# show rib vpn-attributes

Extended community data in RIB:
  Extended community Ref count
  COST:128:128:41984 2
  COST:128:129:42240 2
  COST:128:129:44544 1
  COST:128:129:169984 2
  COST:128:129:307200 1

MVPN attribute data in RIB:
  MVPN Attribute Ref count
  0:0:1:f4:0:0:0:1:1:1:1:1 1
  0:0:2:bc:0:0:0:1:3:3:3:3 10
```
This table describes the significant fields shown in the display.

**Table 94: show rib vpn-attributes Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extended Community</td>
<td>Extended community added by the protocol clients.</td>
</tr>
<tr>
<td>Ref Count</td>
<td>Number of routes referring to the same extended community.</td>
</tr>
<tr>
<td>MVPN Attribute</td>
<td>Connector attribute added by BGP to support MVPNs.</td>
</tr>
<tr>
<td>Ref Count</td>
<td>Number of routes referring to the same extended community.</td>
</tr>
</tbody>
</table>
show route

To display the current routes in the Routing Information Base (RIB), use the `show route` command in XR EXEC mode.

```
show route  [{afi-all|ipv4|ipv6}]  [{unicast|topology topo-name|safi-all}]  [{protocol [instance]ip-address [mask]ip-address/prefix-length}]  [standby]  [detail]
```

**Syntax Description**

- **afi-all** (Optional) Specifies all address families.
- **ipv4** (Optional) Specifies IP Version 4 address prefixes. This is the default.
- **ipv6** (Optional) Specifies IP Version 6 address prefixes.
- **unicast** (Optional) Specifies unicast address prefixes. This is the default.
- **topology topo-name** (Optional) Specifies topology table information and name of the topology table.
- **safi-all** (Optional) Specifies unicast and multicast address prefixes.
- **protocol** (Optional) Name of a routing protocol. If you specify a routing protocol, use one of the following keywords:
  - `bgp`
  - `isis`
  - `ospf`
  - `rip`
  - `static`
  - `local`
  - `connected`
- **instance** (Optional) Number or name used to identify an instance of the specified protocol.
- **ip-address** (Optional) Network IP address about which routing information should be displayed.
- **mask** (Optional) Network mask specified in either of two ways:
  - Network mask can be a four-part, dotted-decimal address. For example, 255.0.0.0 indicates that each bit equal to 1 means the corresponding address bit is a network address.
  - Network mask can be indicated as a slash (/) and number. For example, /8 indicates that the first 8 bits of the mask are 1s, and the corresponding bits of the address are the network address.
- **/prefix-length** (Optional) Length of the IP address prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash (/) must precede the decimal value.
- **standby** (Optional) Displays standby information.
- **detail** (Optional) Displays detailed information for the specified prefix.
Local routes received from RIB are displayed for default ipv4 unicast vrf.

XR EXEC mode

When the `afi-all` keyword is used, the `ip-address` and `mask` arguments are not available.

The following is sample output from the `show route` command when entered without an address:

```
RP/0/RP0/CPU0:router# show route

Codes:  C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
       IA - IS-IS inter area, su - IS-IS summary null, * - candidate default
       U - per-user static route, o - ODR, L - local, G - DAGR
       A - access/subscriber, (!) - FRR Backup path

Gateway of last resort is 1.0.0.1 to network 0.0.0.0

S*  0.0.0.0/0 [1/0] via 1.0.0.1, 13:14:59
C  1.0.0.0/16 is directly connected, 13:14:59, MgmtEth0/5/CPU0/0
L  1.0.14.15/32 is directly connected, 13:14:59, MgmtEth0/5/CPU0/0
C  3.2.3.0/24 is directly connected, 00:04:39, tenGigE 0/3/0/0
L  3.2.3.2/32 is directly connected, 00:04:39, tenGigE 0/3/0/0
O E2 5.2.5.0/24 [110/20] via 3.3.3.1, 00:04:20, tenGigE 0/3/0/0
O E2 6.2.6.0/24 [110/20] via 3.3.3.1, 00:04:20, tenGigE 0/3/0/0
C  7.2.7.0/24 is directly connected, 00:04:20, tenGigE 0/3/0/0
L  7.2.7.2/32 is directly connected, 00:04:20, tenGigE 0/3/0/0
O E2 8.2.8.0/24 [110/20] via 3.3.3.1, 00:04:20, tenGigE 0/3/0/0
C  10.3.0.0/16 is directly connected, 13:14:59, tenGigE 0/0/0/0
L  10.3.0.2/32 is directly connected, 13:14:59, tenGigE 0/0/0/0
```

This table describes the significant fields shown in the display.

Table 95: `show route` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S*</td>
<td>Code indicating how the route was derived. See the code legend preceding the output. In this case, the route was derived from a static (candidate default).</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>[1/0]</td>
<td>First number in the brackets is the administrative distance of the information source; the second number is the metric for the route.</td>
</tr>
<tr>
<td>1.0.0.0/16</td>
<td>Address and prefix length of the remote network.</td>
</tr>
<tr>
<td>MgmtEthernet 0/5/CPU0/0</td>
<td>Specifies the interface through which the specified network can be reached.</td>
</tr>
<tr>
<td>C</td>
<td>Code indicating how the route was derived. See the code legend preceding the output. In this case, the route was connected.</td>
</tr>
<tr>
<td>L</td>
<td>Code indicating how the route was derived. See the code legend preceding the output. In this case, the route was local.</td>
</tr>
<tr>
<td>O</td>
<td>Code indicating how the route was derived. See the code legend preceding the output. In this case, the route was on-demand routing (ODR).</td>
</tr>
<tr>
<td>E2</td>
<td>Code indicating how the route was derived. See the code legend preceding the output. In this case, the route was OSPF external type 2.</td>
</tr>
<tr>
<td>8.2.8.0/24</td>
<td>Address and prefix length of the remote network connected to the static route.</td>
</tr>
<tr>
<td>via 3.3.3.1</td>
<td>Specifies the address of the next router to the remote network.</td>
</tr>
<tr>
<td>13:14:59</td>
<td>Specifies the last time the route was updated.</td>
</tr>
<tr>
<td>(!)</td>
<td>Code indicating fast re-route (FRR) backup path information.</td>
</tr>
</tbody>
</table>

When you specify that you want information about a particular network, more detailed statistics are displayed. The following is sample output from the `show route` command when entered with an IP address:

```
RP/0/RP0/CPU0:router# show route 10.0.0.0

Routing entry for 10.0.0.0/16
  Known via "connected", distance 0, metric 0 (connected)
  Installed Mar 22 22:10:20.906
  Routing Descriptor Blocks
    directly connected, via tenGigE 0/0/0/0
    Route metric is 0
    No advertising protos.
```

Intermediate System-to-Intermediate System (IS-IS) includes an IP address typed length value (TLV) in its link-state packet (LSP) that helps identify the node injecting the route into the network. The IS-IS node uses one of its own interface addresses in this TLV. A loopback address is preferred among interfaces configured under IS-IS. When other networking devices calculate IP routes, they can store the IP address as the originator address with each route in the routing table.

The following example shows the output from the `show route` command for a specific IP address on a router configured with IS-IS. Each path that is shown under the Routing Descriptor Blocks report displays two IP addresses. The first address (10.0.0.9) is the next-hop address; the second is the originator IP address from the advertising IS-IS router.
**RP/0/RP0/CPU0:router# show route 10.0.0.1**

Routing entry for 10.0.0.0/8
Known via "isis", distance 115, metric 10, type level-2
Installed Jan 22 09:26:56.210
Routing Descriptor Blocks:
  * 10.0.0.9, from 10.0.0.9, via tenGigE 2/1
  Route metric is 10
No advertising protos.

This table describes the significant fields shown in the display.

**Table 96: show route with IP Address Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing entry for</td>
<td>Network address and mask.</td>
</tr>
<tr>
<td>Known via</td>
<td>Indicates how the route was derived.</td>
</tr>
<tr>
<td>distance</td>
<td>Administrative distance of the information source.</td>
</tr>
<tr>
<td>metric</td>
<td>Route value assigned by the routing protocol.</td>
</tr>
<tr>
<td>type</td>
<td>IS-IS type level.</td>
</tr>
<tr>
<td>Routing Descriptor Blocks:</td>
<td>Displays the next-hop IP address followed by the information source.</td>
</tr>
<tr>
<td>from ... via ...</td>
<td>First address is the next-hop IP address, and the other is the information source. This report is followed by the interface for this route.</td>
</tr>
<tr>
<td>Route metric</td>
<td>Best metric for this Routing Descriptor Block.</td>
</tr>
<tr>
<td>No advertising protos.</td>
<td>Indicates that no other protocols are advertising the route to their redistribution consumers. If the route is being advertised, protocols are listed in the following manner:</td>
</tr>
</tbody>
</table>

Redist Advertisers:
isis p
ospf 43

The following example illustrates the **show route** command with the **topology topo-name** keyword and argument specified:

**RP/0/RP0/CPU0:router# show route ipv4 multicast topology green**

Codes: C - connected, S - static, R - RIP, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2
       *i - IS-IS inter area, su - IS-IS summary null, * - candidate default
       U - per-user static route, o - ODR, L - local, G - DAGR
       A - access/subscriber, (!) - FRR Backup path

Gateway of last resort is not set
### RIB Commands

**show route**

This example is a sample *show route summary* command output that displays fast-reroute (FRR) Backup path information. The FRR Backup paths are indicated with a (!).

```
RP/0/RP0/CPU0:router# show route summary

Codes:  C - connected, S - static, R - RIP, B - BGP
        O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
        i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
        ia - IS-IS inter area, su - IS-IS summary null, * - candidate default
        U - per-user static route, o - ODR, L - local, G - DAGR
        A - access/subscriber, (!) - FRR Backup path

Gateway of last resort is not set

<table>
<thead>
<tr>
<th>IP Address</th>
<th>Metric</th>
<th>Next Hop</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2.3.4/32</td>
<td>200</td>
<td>via 10.10.1.3</td>
<td>ServiceApp40</td>
</tr>
<tr>
<td>2.0.0.0/30</td>
<td></td>
<td>is directly connected</td>
<td>ServiceApp40</td>
</tr>
<tr>
<td>2.0.1.0/30</td>
<td>13</td>
<td>via 10.10.1.1</td>
<td>ServiceApp43</td>
</tr>
<tr>
<td>2.0.1.1/32</td>
<td>13</td>
<td>via 10.10.1.1</td>
<td>ServiceApp43</td>
</tr>
<tr>
<td>2.4.1.0/24</td>
<td></td>
<td>via 10.10.1.1</td>
<td>ServiceApp43</td>
</tr>
<tr>
<td>2.4.1.2/32</td>
<td>13</td>
<td>via 10.10.1.1</td>
<td>ServiceApp43</td>
</tr>
<tr>
<td>3.1.0.2/32</td>
<td></td>
<td>via 10.10.1.1</td>
<td>ServiceInfra1</td>
</tr>
<tr>
<td>3.1.3.0/30</td>
<td>14</td>
<td>via 10.10.1.1</td>
<td>ServiceInfra2</td>
</tr>
<tr>
<td>3.1.3.2/32</td>
<td>14</td>
<td>via 10.10.1.1</td>
<td>ServiceInfra2</td>
</tr>
<tr>
<td>3.1.3.0/16</td>
<td>20</td>
<td>via 10.10.1.1</td>
<td>ServiceInfra1</td>
</tr>
<tr>
<td>5.3.0.0/16</td>
<td>200</td>
<td>via 10.10.1.1</td>
<td>ServiceInfra1</td>
</tr>
<tr>
<td>5.3.16.10/32</td>
<td>1</td>
<td>via 5.3.16.12</td>
<td>MgmtEth0/RP0/CPU0/0</td>
</tr>
<tr>
<td>5.3.16.12/32</td>
<td>1</td>
<td>via 5.3.16.12</td>
<td>MgmtEth0/RP1/CPU0/0</td>
</tr>
<tr>
<td>5.3.16.16/32</td>
<td>1</td>
<td>via 5.3.16.12</td>
<td>MgmtEth0/RP0/CPU0/0</td>
</tr>
<tr>
<td>5.4.0.0/16</td>
<td>1</td>
<td>via 10.10.1.1</td>
<td>ServiceInfra1</td>
</tr>
<tr>
<td>5.10.0.0/16</td>
<td>1</td>
<td>via 5.3.0.1</td>
<td>tenGigE 0/0/0/3.1 (!)</td>
</tr>
<tr>
<td>O 10.1.1.3/32</td>
<td>110</td>
<td>via 40.1.10.1</td>
<td>Bundle-Ether10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 200.40.1.101</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 100.100.2.1</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td>L 10.1.1.6/32</td>
<td>1</td>
<td>via 10.10.1.1</td>
<td>Loopback0</td>
</tr>
<tr>
<td>O 10.1.1.9/32</td>
<td>110</td>
<td>via 40.1.10.1</td>
<td>Bundle-Ether10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 200.40.1.101</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 100.100.2.1</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td>O 10.1.1.10/32</td>
<td>110</td>
<td>via 40.1.10.1</td>
<td>Bundle-Ether10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 200.40.1.101</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 100.100.2.1</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td>O 10.1.1.11/32</td>
<td>110</td>
<td>via 40.1.11.1</td>
<td>Bundle-Ether1 (!)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 40.3.3.2</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td>O 10.1.1.12/32</td>
<td>110</td>
<td>via 40.1.10.1</td>
<td>Bundle-Ether10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 200.40.1.101</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 100.100.2.1</td>
<td>Bundle-Ether1.1</td>
</tr>
<tr>
<td>O 10.1.1.16/32</td>
<td>110</td>
<td>via 40.1.10.1</td>
<td>Bundle-Ether10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>via 200.40.1.101</td>
<td>Bundle-Ether1.1</td>
</tr>
</tbody>
</table>
```
This example is a sample `show route detail` command output that displays path ID and backup-path ID information:

```
RP/0/RP0/CPU0:router# show route 10.1.1.3 detail
Routing entry for 10.1.1.16/32
Known via "ospf 2", distance 110, metric 21, type intra area
Installed Oct 28 16:07:05.752 for 00:01:56
Routing Descriptor Blocks
  40.1.10.1, from 10.1.1.16, via Bundle-Ether10, Protected
    Route metric is 21
    Label: None
    Tunnel ID: None
    Extended communities count: 0
    Path id:2     Path ref count:0
    Backup path id:33
  200.40.1.101, from 10.1.1.16, via Bundle-Ether1.1, Protected
    Route metric is 21
    Label: None
    Tunnel ID: None
    Extended communities count: 0
    Path id:1     Path ref count:0
    Backup path id:33
  100.100.2.1, from 10.1.1.16, via tenGigE 0/2/0/3.1, Backup
    Route metric is 0
    Label: None
    Tunnel ID: None
    Extended communities count: 0
    Path id:33     Path ref count:2
    Route version is 0xe (14)
    No local label
    IP Precedence: Not Set
    QoS Group ID: Not Set
    Route Priority: RIB_PRIORITY_NON_RECURSIVE_LOW (6) SVD Type RIB_SVD_TYPE_LOCAL
    No advertising protos.
```

This example is a sample `show route ipv6` command output:

```
RP/0/RP0/CPU0:router# show route ipv6
Fri May 18 14:00:10.996 EDT
Codes: C - connected, S - static, R - RIP, B - BGP
O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, su - IS-IS summary null, * - candidate default
U - per-user static route, o - ODR, L - local, G - DAGR
A - access/subscriber, (!) - FRR Backup path
Gateway of last resort is not set
C 1111::2222::abcd/128 is directly connected,
  06:20:02, tenGigE 0/0/0/4
```

This example is a sample `show route ipv6 detail` command output:

```
RP/0/RP0/CPU0:router# show route ipv6 1111:2222::abcd/128 detail
Fri May 18 14:00:20.798 EDT
Routing entry for 1111:2222::abcd/128
Known via "connected l2tpv3_xconnect", distance 0, metric 0 (connected)
Installed May 18 07:40:08.522 for 06:20:12
Routing Descriptor Blocks
```
1111:2222::abcd directly connected, via tenGigE 0/0/0/4
Route metric is 0
Label: 0x2 (2)
Tunnel ID: None
Extended communities count: 0
Route version is 0xd (13)
No local label
IP Precedence: Not Set
QoS Group ID: Not Set
Route Priority: RIB_PRIORITY_CONNECTED (2) SVD Type RIB_SVD_TYPE_LOCAL
Download Priority 0, Download Version 13
No advertising protos.

This example is a sample show route ipv6 summary command output:

RP/0/RP0/CPU0:router#show route ipv6 summary
Fri May 18 14:00:28.988 EDT
Route Source Routes Backup Deleted Memory (bytes)
local 0 0 0 0
connected l2tpv3_xconnect 1 0 0 160
connected 0 0 0 0
Total 1 0 0 160
show route backup

To display backup routes from the Routing Information Base (RIB), use the `show route backup` command in XR EXEC mode.

```
show route [afi-all|ipv4|ipv6] [unicast|topology topo-name|safi-all] backup [ip-address /prefix-length] [standby]
```

### Syntax Description

- **afi-all** (Optional) Specifies all address families.
- **ipv4** (Optional) Specifies IP Version 4 address prefixes.
- **ipv6** (Optional) Specifies IP Version 6 address prefixes.
- **unicast** (Optional) Specifies unicast address prefixes.
- **topology topo-name** (Optional) Specifies topology table information and name of the topology table.
- **safi-all** (Optional) Specifies unicast and multicast address prefixes.
- **ip-address** (Optional) Network IP address about which backup routing information should be displayed.
- **mask** (Optional) Network mask specified in either of two ways:
  - Network mask can be a four-part, dotted decimal address. For example, 255.0.0.0 indicates that each bit equal to 1 means the corresponding address bit is a network address.
  - Network mask can be indicated as a slash (/) and number. For example, /8 indicates that the first 8 bits of the mask are ones, and the corresponding bits of the address are the network address.
- **/prefix-length** (Optional) Length of the IP address prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash (/) must precede the decimal value.
- **standby** (Optional) Displays standby information.

### Command Default

Local routes received from RIB are displayed for default ipv4 unicast vrf.

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `show route backup` command to display information about routes that have been installed into the RIB as backup routes. This command also displays information about the currently selected active route for which there is a backup.

When the **afi-all** keyword is used, the **ip-address** and **mask** arguments are not available.
The following is sample output from the `show route backup` command:

```
RP/0/RP0/CPU0:router# show route backup

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, su - IS-IS summary null, * - candidate default
       U - per-user static route, o - ODR, L - local

S  172.73.51.0/24 is directly connected, 2d20h, tenGigE 4/0/0/1
Backup  O E2 [110/1] via 10.12.12.2, tenGigE 3/0/0/1
```

This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>Code indicating how the route was derived. See the legend of the codes preceding the output.</td>
</tr>
<tr>
<td>172.73.51.0/24</td>
<td>IP address and length of the route.</td>
</tr>
<tr>
<td>2d20h</td>
<td>Time (in hh:mm:ss) since the route was installed in the RIB.</td>
</tr>
<tr>
<td>tenGigE4/0/0/1</td>
<td>Outbound interface for the route.</td>
</tr>
<tr>
<td>Backup</td>
<td>Identifies the entry as a backup version of the route, typically installed by a different routing protocol.</td>
</tr>
<tr>
<td>O</td>
<td>Code indicating how the route was derived. See the code legend preceding the output.</td>
</tr>
</tbody>
</table>
### Field | Description
--- | ---
E2 | Code for the type of route. This code is relevant only for OSPF and IS-IS routes.
 | The codes for an OSPF route can be:
 | none—intra-area route
 | IA—interarea route
 | E1—external type 1
 | E2—external type 2
 | N1—NSSA external type 1
 | N2—NSSA external type 2
 | The codes for an IS-IS route can be:
 | L1—level 1
 | L2—level 2
 | ia—interarea
 | su—summary route

[110/1] | Distance and metric for the route.
10.12.12.2 | IP address of next-hop on the route.
tenGigE 3/0/0/1 | Outbound interface for the OSPF version of the route.
show route best-local

To display the best local address to use for return packets from the given destination, use the `show route best-local` command in XR EXEC mode.

```
show route [{ipv4|ipv6}] [{unicast|topology topo-name|safi-all}] best-local ip-address | standby
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies IP Version 4 address prefixes.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td>unicast</td>
<td>(Optional) Specifies unicast address prefixes.</td>
</tr>
<tr>
<td>topology topo-name</td>
<td>(Optional) Specifies topology table information and name of the topology table.</td>
</tr>
<tr>
<td>safi-all</td>
<td>(Optional) Specifies unicast and multicast address prefixes.</td>
</tr>
<tr>
<td>ip-address</td>
<td>IP address about which best local information should be displayed.</td>
</tr>
<tr>
<td>standby</td>
<td>(Optional) Displays standby information.</td>
</tr>
</tbody>
</table>

**Command Default**

Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show route best-local` command to display information about the best local routes in the routing table.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show route best-local` command:

```
RP/0/RP0/CPU0:router# show route best-local 10.12.12.1/32

Routing entry for 10.12.12.1/32
   Known via "local", distance 0, metric 0 (connected)
Routing Descriptor Blocks
   10.12.12.1 directly connected, via tenGigE 3/0/0/1
   Route metric is 0
```
This table describes the significant fields shown in the display.

**Table 98: show route best-local Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Routing entry for</td>
<td>Identifies the requested IP address.</td>
</tr>
<tr>
<td>Known via</td>
<td>Indicates how the route was derived.</td>
</tr>
<tr>
<td>distance</td>
<td>Administrative distance of the information source.</td>
</tr>
<tr>
<td>metric</td>
<td>Route value assigned by the routing protocol.</td>
</tr>
<tr>
<td>Routing Descriptor Blocks:</td>
<td>Displays the next-hop IP address followed by the information source.</td>
</tr>
<tr>
<td>10.12.12.1 Directly connected ... via ...</td>
<td>First address is the next-hop IP address, followed by a report that it is directly connected. This report is followed by the interface for this route.</td>
</tr>
</tbody>
</table>
show route connected

To display the current connected routes of the routing table, use the `show route connected` command in XR EXEC mode.

```
show route [{afi-all|ipv4|ipv6}] [{unicast|topology topo-name|safi-all}] connected [ standby]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>afi-all</code></td>
<td>(Optional) Specifies all address families.</td>
</tr>
<tr>
<td><code>ipv4</code></td>
<td>(Optional) Specifies IP Version 4 address prefixes.</td>
</tr>
<tr>
<td><code>ipv6</code></td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td><code>unicast</code></td>
<td>(Optional) Specifies unicast address prefixes.</td>
</tr>
<tr>
<td><code>topology topo-name</code></td>
<td>(Optional) Specifies topology table information and name of the topology table.</td>
</tr>
<tr>
<td><code>safi-all</code></td>
<td>(Optional) Specifies unicast and multicast address prefixes.</td>
</tr>
<tr>
<td><code>standby</code></td>
<td>(Optional) Displays standby information.</td>
</tr>
</tbody>
</table>

### Command Default

Local routes received from RIB are displayed for default ipv4 unicast vrf.

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `show route connected` command to display information about connected routes in the routing table.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

### Examples

The following is sample output from the `show route connected` command:

```
RP/0/RP0/CPU0:router# show route connected
C  1.68.0.0/16 is directly connected, 13:43:40, MgmtEth0/5/CPU0/0
C  3.3.3.0/24 is directly connected, 00:23:23, tenGigE 0/3/0/0
C  7.7.7.0/24 is directly connected, 00:33:00, tenGigE 0/3/0/7
C  10.0.0.0/16 is directly connected, 13:43:40, tenGigE 0/0/0/0
C  10.10.10.0/30 is directly connected, 13:43:40, Loopback0
C  11.11.11.0/24 is directly connected, 13:43:40, Loopback11
```
This table describes the significant fields shown in the display.

**Table 99: show route connected Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>Code to indicate the route is connected.</td>
</tr>
<tr>
<td>1.68.0.0/16</td>
<td>IP address and length of the route.</td>
</tr>
<tr>
<td>13:43:40</td>
<td>Time (in hh:mm:ss) since the route was installed in the RIB.</td>
</tr>
<tr>
<td>MgmtEth0/5/CPU0/0</td>
<td>Outbound interface for the route.</td>
</tr>
</tbody>
</table>
show route local

To display local routes receiving routing updates from the Routing Information Base (RIB), use the show route local command in XR EXEC mode.

```
show route [afi-all|ipv4|ipv6] [unicast|topology topo-name|safi-all] local [type interface-path-id] [standby]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>afi-all</td>
<td>(Optional) Specifies all address families.</td>
</tr>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies IP Version 4 address prefixes.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td>unicast</td>
<td>(Optional) Specifies unicast address prefixes.</td>
</tr>
<tr>
<td>topology topo-name</td>
<td>(Optional) Specifies topology table information and name of the topology table.</td>
</tr>
<tr>
<td>safi-all</td>
<td>(Optional) Specifies unicast and multicast address prefixes.</td>
</tr>
<tr>
<td>type</td>
<td>Interface type. For more information, use the question mark ( ? ) online help function.</td>
</tr>
<tr>
<td>interface-path-id</td>
<td>Physical interface or virtual interface.</td>
</tr>
</tbody>
</table>

**Note**

Use the `show interfaces` command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark ( ? ) online help function.

### Command Default

Local routes received from RIB are displayed for default ipv4 unicast vrf.

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `show route local` command to display information about local routes in the routing table.

### Task ID

- **Task ID**
  - rib : read

### Examples

The following is sample output from the `show route local` command:
RP/0/RP0/CPU0:router# show route local

L 10.10.10.1/32 is directly connected, 00:14:36, Loopback0
L 10.91.36.98/32 is directly connected, 00:14:32, tenGigE6/0/0/1
L 172.22.12.1/32 is directly connected, 00:13:35, tenGigE3/0/0/1
L 192.168.20.2/32 is directly connected, 00:13:27, tenGigE4/0/0/1
L 10.254.254.1/32 is directly connected, 00:13:26, tenGigE5/0/0/1

This table describes the significant fields shown in the display.

Table 100: show route local Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>L</td>
<td>Code to indicate the route is local.</td>
</tr>
<tr>
<td>10.10.10.1/32</td>
<td>IP address and length of the route.</td>
</tr>
<tr>
<td>00:14:36</td>
<td>Time (in hh:mm:ss) since the route was installed in the RIB.</td>
</tr>
<tr>
<td>Loopback0</td>
<td>Outbound interface for the route.</td>
</tr>
</tbody>
</table>
show route longer-prefixes

To display the current routes in the Routing Information Base (RIB) that share a given number of bits with a given network, use the `show route longer-prefixes` command in XR EXEC mode.

```
show route [ipv4|ipv6] [unicast|topology topo-name|safi-all] longer-prefixes {ip-address mask ip-address/prefix-length} [standby]
```

**Syntax Description**

- `ipv4` (Optional) Specifies IP Version 4 address prefixes.
- `ipv6` (Optional) Specifies IP Version 6 address prefixes.
- `unicast` (Optional) Specifies unicast address prefixes.
- `topology topo-name` (Optional) Specifies topology table information and name of the topology table.
- `safi-all` (Optional) Specifies unicast and multicast address prefixes.
- `ip-address` Network IP address about which routing information should be displayed.
- `mask` Network mask specified in either of two ways:
  - Network mask can be a four-part, dotted-decimal address. For example, 255.0.0.0 indicates that each bit equal to 1 means the corresponding address bit is a network address.
  - Network mask can be indicated as a slash (/) and number. For example, /8 indicates that the first 8 bits of the mask are 1s, and the corresponding bits of the address are the network address.
- `/prefix-length` Length of the IP address prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash (/) must precede the decimal value.
- `standby` (Optional) Displays standby information.

**Command Default**

Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show route longer-prefixes` command to troubleshoot forwarding problems whose cause may be a long prefix.

**Task ID**

<table>
<thead>
<tr>
<th>Task</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>
The following is sample output from the `show route longer-prefixes` command:

```
RP/0/RP0/CPU0:router# show route longer-prefixes 172.16.0.0/8

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
       IA - IS-IS inter area, su - IS-IS summary null, * - candidate default
       U - per-user static route, o - ODR, L - local

L  172.29.52.70/32 is directly connected, 4d15h, MgmtEth0/RSP0RP0/CPU0/0
L  172.29.52.71/32 is directly connected, 4d15h, MgmtEth0/RP1/CPU0/0
L  172.29.52.72/32 [0/0] via 172.29.52.72, 4d15h, MgmtEth0/RSP0RP0/CPU0/0
```

This table describes the significant fields shown in the display.

### Table 101: `show route longer-prefixes` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>172.29.52.70/32</td>
<td>IP address and length of the route.</td>
</tr>
<tr>
<td>4d15h</td>
<td>Time (in hh:mm:ss or ndnh) since the route was installed in the RIB.</td>
</tr>
<tr>
<td>MgmtEth0/RSP0 RP0/CPU0/0</td>
<td>Outbound interface for the route.</td>
</tr>
</tbody>
</table>
show route next-hop

To filter routes by the next-hop address or interface, use the `show route next-hop` command in XR EXEC mode.

```
show route [{ipv4|ipv6}] [{unicast|topology topo-name|safi-all}] next-hop [ip-address]|[{standby}]
```

**Syntax Description**

- `ipv4` (Optional) Specifies IP Version 4 address prefixes.
- `ipv6` (Optional) Specifies IP Version 6 address prefixes.
- `unicast` (Optional) Specifies unicast address prefixes.
- `topology topo-name` (Optional) Specifies topology table information and name of the topology table.
- `safi-all` (Optional) Specifies unicast and multicast address prefixes.
- `ip-address` (Optional) IP address about which next-hop information is to be displayed.
- `standby` (Optional) Displays standby information.

**Command Default**

Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show route next-hop` command to find all routes going through a next-hop address or interface.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show route next-hop` command filtering routes on the next-hop address:

```
RP/0/RP0/CPU0:router# show route next-hop 1.68.0.1
Codes:  C - connected,  S - static,  R - RIP,  M - mobile,  B - BGP
        O - OSPF,  IA - OSPF inter area
        N1 - OSPF NSSA external type 1,  N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1,  E2 - OSPF external type 2,  E - EGP
        i - ISIS,  L1 - IS-IS level-1,  L2 - IS-IS level-2
        * - IS-IS inter area,  su - IS-IS summary null,  * - candidate default
        U - per-user static route,  o - ODR,  L - local
```
Gateway of last resort is 1.68.0.1 to network 0.0.0.0

S* 0.0.0.0/0 [1/0] via 1.68.0.1, 15:01:49
S 223.255.254.254/32 [1/0] via 1.68.0.1, 15:01:49

The following is sample output from the `show route next-hop` command filtering routes on the next-hop interface:

```
RP/0/RP0/CPU0:router# show route next-hop tenGigE 0/1/0/2

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, su - IS-IS summary null, * - candidate default
       U - per-user static route, o - ODR, L - local

Gateway of last resort is 1.68.0.1 to network 0.0.0.0

C 11.1.1.0/24 is directly connected, 15:01:46, tenGigE 0/1/0/2
L 11.1.1.2/32 is directly connected, 15:01:46, tenGigE 0/1/0/2
```

This table describes the significant fields shown in the display.

**Table 102: show route next-hop Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.1.1.0/24</td>
<td>IP address and length of the route.</td>
</tr>
<tr>
<td>15:01:46</td>
<td>Time (in hh:mm:ss or ndnh) since the route was installed in the RIB.</td>
</tr>
<tr>
<td>tenGigE0/1/0/2</td>
<td>Outbound interface for the route.</td>
</tr>
</tbody>
</table>
show route quarantined

To display mutually recursive (looping) routes, use the `show route quarantined` command in XR EXEC mode.

```
show route [{ipv4|ipv6}] [{unicast|topology topo-name}|safi-all] quarantined [{ip-address/prefix-length}|ip-address mask] [standby]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies IP Version 4 address prefixes.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td>unicast</td>
<td>(Optional) Specifies unicast address prefixes.</td>
</tr>
<tr>
<td>topology</td>
<td>(Optional) Specifies topology table information and name of the topology table.</td>
</tr>
<tr>
<td>topo-name</td>
<td>(Optional) Specifies unicast and multicast address prefixes.</td>
</tr>
<tr>
<td>safi-all</td>
<td>(Optional) Specifies unicast address prefix information.</td>
</tr>
<tr>
<td>ip-address</td>
<td>(Optional) IP address about which looping routes information is to be displayed.</td>
</tr>
<tr>
<td>/ prefix-length</td>
<td>(Optional) Length of the IP address prefix. A decimal value that indicates how many of the high-order contiguous bits of the address compose the prefix (the network portion of the address). A slash (/) must precede the decimal value.</td>
</tr>
<tr>
<td>ip-address mask</td>
<td>(Optional) Network mask applied to the <code>ip-address</code> argument.</td>
</tr>
<tr>
<td>standby</td>
<td>(Optional) Displays standby information.</td>
</tr>
</tbody>
</table>

**Command Default**

Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

RIB quarantining detects mutually recursive routes and quarantines the last route that actually completes the mutual recursion. The quarantined route is periodically evaluated to see if the mutual recursion has gone away. If the recursion still exists, the route remains quarantined. If the recursion has gone away, the route is released from quarantine.

Use the `show route quarantined` command to display mutually recursive (looping) routes.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show route quarantined` command:
RP/0/RP0/CPU0:router# show route quarantined

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - ISIS, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, su - IS-IS summary null, * - candidate default
U - per-user static route, o - ODR, L - local

S 10.10.109.1/32 [1/0] via 10.10.34.1, 00:00:01 (quarantined)
  [1/0] via 10.10.37.1, 00:00:01 (quarantined)
  [1/0] via 10.10.60.1, 00:00:01 (quarantined)
  [1/0] via 10.10.68.1, 00:00:01 (quarantined)
  [1/0] via 10.10.91.1, 00:00:01 (quarantined)
  [1/0] via 10.10.93.1, 00:00:01 (quarantined)
  [1/0] via 10.10.97.1, 00:00:01 (quarantined)
S 10.0.0.0/8 [1/0] via 11.11.11.11, 00:01:29 (quarantined)
S 10.10.0.0/16 [1/0] via 11.11.11.11, 00:01:29 (quarantined)
S 10.10.10.0/24 [1/0] via 11.11.11.11, 00:01:29 (quarantined)
S 10.10.10.10/32 [1/0] via 11.11.11.11, 00:00:09 (quarantined)

This table describes the significant fields shown in the display.

Table 103: show route quarantined Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.10.109.1/32</td>
<td>IP address and length of the route.</td>
</tr>
<tr>
<td>[1/0]</td>
<td>Distance and metric for the route.</td>
</tr>
<tr>
<td>via 10.10.34.1</td>
<td>IP address of next-hop on the route.</td>
</tr>
<tr>
<td>00:00:01</td>
<td>Time (in hh:mm:ss or n unh) since the route was installed in the RIB.</td>
</tr>
<tr>
<td>(quarantined)</td>
<td>Shows that the route is quarantined.</td>
</tr>
</tbody>
</table>
show route resolving-next-hop

To display the next-hop gateway or host to a destination address, use the show route resolving-next-hop command in XR EXEC mode.

show route  [{ipv4|ipv6}]  [{unicast|topology  topo-name}|safi-all]}  resolving-next-hop  ip-address [ standby]

Syntax Description

- **ipv4** (Optional) Specifies IP Version 4 address prefixes.
- **ipv6** (Optional) Specifies IP Version 6 address prefixes.
- **unicast** (Optional) Specifies unicast address prefixes.
- **topology  topo-name** (Optional) Specifies topology table information and name of the topology table.
- **safi-all** (Optional) Specifies unicast and multicast address prefixes.
- **ip-address** IP address about which resolved next-hop information is to be displayed.
- **standby** (Optional) Displays standby information.

Command Default

Local routes received from RIB are displayed for default ipv4 unicast vrf.

Command Modes

XR EXEC mode

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| Release 6.0 | This command was introduced.

Usage Guidelines

Use the show route resolving-next-hop command to perform a recursive route lookup on the supplied destination address and return information on the next immediate router (next hop) to the destination.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

Examples

The following is sample output from the show route resolving-next-hop command:

```
RP/0/RP0/CPU0:router# show route resolving-next-hop 10.1.1.1

Nexthop matches 10.1.1.1/32
   Known via "local", distance 0, metric 0 (connected)
   Installed Aug 22 01:57:08.514
   Directly connected nexthops
      10.1.1.1 directly connected, via Loopback0
   Route metric is 0
```
This table describes the significant fields shown in the display.

**Table 104: show route resolving-next-hop Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Known via</td>
<td>Name of the routing protocol that installed the matching route.</td>
</tr>
<tr>
<td>Route metric is</td>
<td>Metric of the route.</td>
</tr>
</tbody>
</table>
show route static

To display the current static routes of the Routing Information Base (RIB), use the `show route static` command in XR EXEC mode.

```
show route  [{afi-all|ipv4|ipv6}]  [{unicast|topology topo-name|safi-all}]  static  [ standby]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>afi-all</td>
<td>(Optional) Specifies all address families.</td>
</tr>
<tr>
<td>ipv4</td>
<td>(Optional) Specifies IP Version 4 address prefixes.</td>
</tr>
<tr>
<td>ipv6</td>
<td>(Optional) Specifies IP Version 6 address prefixes.</td>
</tr>
<tr>
<td>unicast</td>
<td>(Optional) Specifies unicast address prefixes.</td>
</tr>
<tr>
<td>topology topo-name</td>
<td>(Optional) Specifies topology table information and name of the topology table.</td>
</tr>
<tr>
<td>safi-all</td>
<td>(Optional) Specifies unicast and multicast address prefixes.</td>
</tr>
<tr>
<td>standby</td>
<td>(Optional) Displays standby information.</td>
</tr>
</tbody>
</table>

**Command Default**

Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show route static` command to display information about static routes in the routing table.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>rib</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output from the `show route static` command:

```
RP/0/RP0/CPU0:router# show route static
S  10.1.1.0/24 is directly connected, 00:54:05, tenGigE3/0/0/1
```
This table describes the significant fields shown in the display.

**Table 105: show route static Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>S</td>
<td>Code to indicate the route is static.</td>
</tr>
<tr>
<td>10.1.1.0/24</td>
<td>IP address and distance for the route.</td>
</tr>
<tr>
<td>00:54:05</td>
<td>Time (in hh:mm:ss) since the route was installed in the RIB.</td>
</tr>
<tr>
<td>tenGigE3/0/0/1</td>
<td>Outbound interface for the route.</td>
</tr>
<tr>
<td>[1/0]</td>
<td>Distance and metric for the route.</td>
</tr>
</tbody>
</table>
show route summary

To display the current contents of the Routing Information Base (RIB), use the `show route summary` command in XR EXEC mode mode.

```
show route [afi-all|ipv4|ipv6] [unicast|topology topo-name|safi-all] summary [detail] [standby]
```

**Syntax Description**

- **afi-all** (Optional) Specifies all address families.
- **ipv4** (Optional) Specifies IP Version 4 address prefixes.
- **ipv6** (Optional) Specifies IP Version 6 address prefixes.
- **unicast** (Optional) Specifies unicast address prefixes.
- **topology topo-name** (Optional) Specifies topology table information and name of the topology table.
- **safi-all** (Optional) Specifies unicast and multicast address prefixes.
- **detail** (Optional) Displays a detailed summary of the contents of the RIB, including the number of paths and some protocol-specific route attributes.
- **standby** (Optional) Displays standby information.

**Command Default**

Local routes received from RIB are displayed for default ipv4 unicast vrf.

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show route summary` command to display information about routes in the routing information base. When a route summary is needed frequently—for instance, in a polling situation—use the `show route summary` command without the `detail` keyword. The `detail` keyword is used less frequently for verification purposes, because it is much more expensive (in bandwidth), requiring a scan of the entire routing database.

**Task ID**

- **Task ID**
  - rib
  - read

**Examples**

The following is sample output from the `show route summary` command:

```
RP/0/RP0/CPU0:router# show route summary
Route Source Routes Backup Deleted Memory (bytes)
```
This table explains fields in the output of the `show route summary` command.

### Table 106: show route summary Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route Source</td>
<td>Routing protocol name.</td>
</tr>
<tr>
<td>Routes</td>
<td>Number of selected routes that are present in the routing table for each route source.</td>
</tr>
<tr>
<td>Backup</td>
<td>Number of routes that are not selected (are backup to a selected route).</td>
</tr>
<tr>
<td>Deleted</td>
<td>Number of routes that have been marked for deletion in the RIB, but have not yet been purged.</td>
</tr>
<tr>
<td>Memory</td>
<td>Number of bytes allocated to maintain all routes for the particular route source.</td>
</tr>
</tbody>
</table>

The following is sample output from the `show route summary` command with the `detail` keyword:

```
RP/0/RP0/CPU0:router# show route summary detail

        Route Source       Active Route Active Path Backup Route Backup Path
       ------------        ------------  -----------  -----------  -----------
         static          1            1          0           0          0
         connected       2            2          1           1          1
         local           3            3          0           0          0
         isis            1            1          1           1          1
       Level 1:          0            0          1           1          1
       Level 2:          1            1          0           0          0
         ospf 1673       6            12          0           0          0
      Intra-Area:       3            6          0           0          0
     Inter-Area:       3            6          0           0          0
      External-1:      0            0           0           0          0
      External-2:      0            0           0           0          0
         bgp 100        10           20          4           8           8
     Externals:        5            10          4           8           8
        Internal:      5            10          0           0          0
        local:         0            0           0           0          0
        Total         7             7           2           2          2
```

This table explains fields in the output of the `show route summary detail` command.

### Table 107: show route summary detail Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route Source</td>
<td>Source of the route. Routing protocol name and type.</td>
</tr>
<tr>
<td>Active Route</td>
<td>Number of active routes present in the routing table for each route source.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Active Path</td>
<td>Number of active paths present in the routing table for each route source.</td>
</tr>
<tr>
<td>Backup Route</td>
<td>Number of routes that are backup to a selected route for each route source.</td>
</tr>
<tr>
<td>Backup Path</td>
<td>Number of paths that are backup to a selected path for each route source.</td>
</tr>
</tbody>
</table>
show route summary
Routing Policy Language Commands

This module describes the Cisco IOS XR software routing policy language (RPL) commands used to create, modify, monitor, and maintain routing policies.

For detailed information about RPL concepts, configuration tasks, and examples, see the Implementing Routing Policy on Cisco NCS 5000 Series Routers module in the Routing Configuration Guide for Cisco NCS 5000 Series Routers.

Note
Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

- abort (RPL), on page 547
- add, on page 549
- apply, on page 550
- as-path in, on page 552
- as-path is-local, on page 554
- as-path length, on page 555
- as-path neighbor-is, on page 556
- as-path originates-from, on page 558
- as-path passes-through, on page 560
- as-path-set, on page 562
- as-path unique-length, on page 564
- community is-empty, on page 566
- community matches-any, on page 567
- community matches-every, on page 569
- community-set, on page 571
- delete community, on page 574
- delete extcommunity rt, on page 576
- destination in, on page 578
- done, on page 580
- drop, on page 582
- edit, on page 583
- end-global, on page 586
- end-policy, on page 587
• end-set, on page 588
• extcommunity rt is-empty, on page 590
• extcommunity rt matches-any, on page 591
• extcommunity rt matches-every, on page 593
• extcommunity rt matches-within, on page 595
• extcommunity-set cost, on page 597
• extcommunity-set rt, on page 599
• extcommunity-set soo, on page 601
• extcommunity soo is-empty, on page 603
• extcommunity soo matches-any, on page 604
• extcommunity soo matches-every, on page 606
• if, on page 608
  • if route-aggregated, on page 614
• is-best-path, on page 615
• is-backup-path, on page 616
• is-multi-path, on page 617
• local-preference, on page 618
• med, on page 619
• next-hop in, on page 620
• orf prefix in, on page 621
• origin is, on page 623
• ospf-area, on page 625
• pass, on page 627
• path-type is, on page 628
• policy-global, on page 629
• prefix-set, on page 630
• prepend as-path, on page 632
• protocol, on page 634
• rd in, on page 636
• rd-set, on page 637
• replace as-path, on page 639
• remove as-path private-as, on page 641
• rib-has-route, on page 642
• route-has-label, on page 643
• route-policy (RPL), on page 644
• route-type is, on page 646
• rpl editor, on page 648
• rpl maximum, on page 649
• rpl set-exit-as-abort, on page 650
• set administrative-distance, on page 651
• set aigp-metric, on page 652
• set community, on page 653
• set core-tree, on page 655
• set dampening, on page 656
• set extcommunity cost, on page 658
• set extcommunity rt, on page 660
• set ip-precedence, on page 662
• set isis-metric, on page 663
• set label, on page 664
• set label-mode, on page 665
• set level, on page 667
• set local-preference, on page 668
• set med, on page 669
• set metric-type (IS-IS), on page 671
• set metric-type (OSPF), on page 673
• set next-hop, on page 674
• set origin, on page 676
• set ospf-metric, on page 677
• set path-selection, on page 678
• set qos-group (RPL), on page 679
• set rib-metric, on page 680
• set rip-metric, on page 681
• set rip-tag, on page 682
• set rpf-topology, on page 683
• set spf-priority, on page 685
• set tag, on page 686
• set traffic-index, on page 687
• set vpn-distinguisher, on page 689
• set weight, on page 690
• show rpl, on page 691
• show rpl active as-path-set, on page 693
• show rpl active community-set, on page 695
• show rpl active extcommunity-set, on page 697
• show rpl active prefix-set, on page 700
• show rpl active rd-set, on page 702
• show rpl active route-policy, on page 704
• show rpl as-path-set, on page 706
• show rpl as-path-set attachpoints, on page 707
• show rpl as-path-set references, on page 709
• show rpl community-set, on page 711
• show rpl community-set attachpoints, on page 713
• show rpl community-set references, on page 715
• show rpl extcommunity-set, on page 717
• show rpl inactive as-path-set, on page 720
• show rpl inactive community-set, on page 722
• show rpl inactive extcommunity-set, on page 724
• show rpl inactive prefix-set, on page 726
• show rpl inactive rd-set, on page 728
• show rpl inactive route-policy, on page 730
• show rpl maximum, on page 732
• show rpl policy-global references, on page 734
• show rpl prefix-set, on page 736
• show rpl prefix-set attachpoints, on page 737
• show rpl prefix-set references, on page 739
• show rpl rd-set, on page 741
• show rpl rd-set attachpoints, on page 742
• show rpl rd-set references, on page 744
• show rpl route-policy, on page 746
• show rpl route-policy attachpoints, on page 749
• show rpl route-policy inline, on page 751
• show rpl route-policy references, on page 753
• show rpl route-policy uses, on page 756
• show rpl unused as-path-set, on page 759
• show rpl unused community-set, on page 762
• show rpl unused extcommunity-set, on page 765
• show rpl unused prefix-set, on page 766
• show rpl unused rd-set, on page 769
• show rpl unused route-policy, on page 770
• source in, on page 773
• suppress-route, on page 775
• tag, on page 776
• tag in, on page 777
• tag-set, on page 779
• unsuppress-route, on page 780
• vpn-distinguisher is, on page 781
**abort (RPL)**

To discard a route policy or set definition and return to XR Config mode, use the `abort` command in the appropriate configuration mode.

```
abort
```

**Syntax Description**

This command has no keywords or arguments.

**Command Default**

No default behavior or values

**Command Modes**

- Route-policy configuration
- Prefix set configuration
- Route distinguisher set configuration
- AS path set configuration
- Community set configuration
- Extended community set configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to discard a route policy definition that was started and return to XR Config mode:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# if as-path is-local then
RP/0/RP0/CPU0:router(config-rpl-if)# abort
RP/0/RP0/CPU0:router(config)#
```

The following example shows how to discard a prefix set definition that was started and return to XR Config mode:

```
RP/0/RP0/CPU0:router(config)# prefix-set legal-ipv4-prefix-examples
RP/0/RP0/CPU0:router(config-pfx)# 10.0.1.1,
RP/0/RP0/CPU0:router(config-pfx)# 10.0.2.0/24,
```
abort (RPL)

RP/0/RP0/CPU0:router(config-pfx)# abort
RP/0/RP0/CPU0:router(config)#
add

To add a value to an Routing Information Protocol (RIP) existing metric, use the `add` command in route-policy configuration mode.

```
add {rip-metric {numberparameter}}
```

**Syntax Description**
- **rip-metric** Specifies an RIP metric attribute.
- **number** Value assigned to a four-bit unsigned integer. Range is from 0 to 16.
- **parameter** Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
If the add value is greater than the maximum allowed value, the metric is added. If the resulting metric exceeds the maximum for the routing protocol, then the route is dropped (by the client routing protocol).

**Task ID**
- **Task ID**: route-policy
- **Operations**: read, write

**Examples**
The following example shows how to offset the RIP metric value:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# add rip-metric 4
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**apply**

To execute a parameterized or unparameterized policy from within another policy, use the `apply` command in route-policy configuration mode.

```
apply policy_name [ { argument1, argument2, ..., argumentN } ]
```

**Syntax Description**

- `policy_name` Name of a route policy.
- `argument` (Optional) Parameter name. The `argument` can be a value (for example, '100') or a parameter (for example, 'Sparameter')

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

Use the `apply` command to execute a policy (either parameterized or unparameterized) from within another policy, which allows for the reuse of common blocks of policy.

Wildcards can be used in apply policy names. This supports the nested wildcard apply scenario. A wildcard is specified by inserting an asterisk (*) in place of one of the portions of the apply policy name; the wildcard indicates that any value for that portion of the apply policy name matches. The nested wildcard apply policy allows wildcard (*) based apply nesting. The wildcard operation permits declaration of a generic apply statement that calls all policies that contain a specific defined set of alphanumeric characters, defined on the router.

**Examples**

In the following example, the policy CustomerIn applies the route-policy SetLocalPref to conditionally set the local preference on a route. The parameters 20, 30, 40, and 50 are passed to the parameterized policy SetLocalPref, where the local preference is set to:

- 20, if the community 217:20 is present in the route
- 30, if the community 217:30 is present in the route
- 40, if the community 217:40 is present in the route
- 50, if the community 217:50 is present in the route

```
RP/0/RP0/CPU0:router(config)# route-policy SetLocalPref ($lp0, $lp1, $lp2, $lp3, $lp4)
RP/0/RP0/CPU0:router(config-rpl)# if community matches-any ($lp0:$lp1)then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference $lp1
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif community matches-any ($lp0:$lp2) then
```
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference $lp2
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif community matches-any ($lp0:$lp3) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference $lp3
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif community matches-any ($lp0:$lp4) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference $lp4
RP/0/RP0/CPU0:router(config-rpl-elseif)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy

RP/0/RP0/CPU0:router(config)# route-policy CustomerIn($cust)
RP/0/RP0/CPU0:router(config-rpl)# apply SetLocalPref ($cust, 20, 30, 40, 50)
RP/0/RP0/CPU0:router(config-rpl)# end-policy

RP/0/RP0/CPU0:router(config)# route-policy Cust_217
RP/0/RP0/CPU0:router(config-rpl)# apply CustomerIn(217)
RP/0/RP0/CPU0:router(config-rpl)# end-policy
To match the AS path of a route to an AS path set, use the **as-path in** command in route-policy configuration mode.

**Syntax**

```
as-path in {as-path-set-name inline-as-path-set parameter}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>as-path-set-name</td>
<td>Name of an AS path set.</td>
</tr>
<tr>
<td>inline-as-path-set</td>
<td>Inline AS path set. The inline AS path set must be enclosed in parentheses.</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **as-path in** command as a conditional expression within an **if** statement to match the AS path of a route to an AS path set. The AS path is a sequence of autonomous system numbers traversed by a route.

For a list of all conditional expressions available within an **if** statement, see the **if** command.

The **as-path in** command evaluates to true if at least one of the regular expressions defined in the associated AS path set matches the AS path attribute of the route.

In the case where the AS path set is defined but contains no elements in it, the **as-path in** conditional expression command returns false.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read,</td>
</tr>
<tr>
<td></td>
<td>write</td>
</tr>
</tbody>
</table>

**Examples**

For example, assume we have an AS path set named my-as-set defined as follows:

```
RP/0/RP0/CPU0:router(config)# as-path-set my-as-set
RP/0/RP0/CPU0:router(config-as)# ios-regex '_'12$' ,
RP/0/RP0/CPU0:router(config-as)# ios-regex '_'13$'
RP/0/RP0/CPU0:router(config-as)# end-set
```

and the following policy excerpt using an **as-path-set-name** argument:
The AS path in condition evaluates to true if one or more of the regular expression matches in the set my-as-set match the AS path associated with the route. In the case of a defined but empty AS path set, this operator returns false.

The preceding policy excerpt is equivalent to the following version, which uses an *inline-as-path* set variable:

```cpp
RP/0/RP0/CPU0:router(config-rpl)# if as-path in (ios-regex '_12$',ios-regex '_13$') then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
as-path is-local

To determine if this router or another router within this autonomous system or confederation originated a Border Gateway Protocol (BGP) route, use the `as-path is-local` command in route-policy configuration mode.

**as-path is-local**

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `as-path is-local` command as a conditional expression within an `if` statement to determine if this router (or another router within this autonomous system or confederation) originated the route.

**Note**

For a list of all conditional expressions available within an `if` statement, see the `if` command.

Routes that are locally originated within the autonomous system or confederation carry an empty AS path. For the Border Gateway Protocol (BGP) specification, when a route is advertised across the autonomous system boundary or a confederation boundary, the local autonomous system number or confederation ID is appended to the autonomous system path. The AS path of a locally originated aggregate is also empty unless it has been modified by policy.

The `is-local` operator evaluates to true for autonomous system paths that are empty. An empty AS path is how an AS path that is local to our autonomous system is represented in BGP.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, if the AS path is local, then the local preference is set to 100:

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path is-local then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
as-path length

To compare the number of ASN in the AS path of a Border Gateway Protocol (BGP) route, use the `as-path length` command in route-policy configuration mode.

```
as-path length \{eq|is|ge|le\} \{number\}
```

### Syntax Description

- `eq` | `is` | `ge` | `le`  
  Equal to; greater than or equal to; less than or equal to.

- `number`  
  Value assigned to an 11-bit unsigned integer. Range is from 0 to 2047.

- `parameter`  
  Parameter name. The parameter name must be preceded with a “$.”

### Command Default

No default behavior or values

### Command Modes

Route-policy configuration

### Command History

**Release** | **Modification**
--- | ---
Release 6.0 | This command was introduced.

### Usage Guidelines

Use the `as-path length` command as a conditional expression within an `if` statement to perform a conditional check based on the length of the AS path.

For a list of all conditional expressions available within an `if` statement, see the `if` command.

This command takes either a specific integer value or a range of integer values specified with the `ge` and `le` operators. Any or all these integers can be parameterized. The operator counts one for each autonomous system in the path. In the case where the route may be aggregated and contain one or more AS sets, the length operator adds one for each set present, the occurrence of an AS set typically indicates that this route is an aggregated route, and the aggregated route had a component route that contained one of the autonomous systems in the set. Likewise, in the case of confederations, a count of one is added for each confederation in the path or each confederation set in the path. A null AS path has a length of zero.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

In the following example, if the AS path length equals 10, then the local preference is set to 100:

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path length eq 10 then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
as-path neighbor-is

To test autonomous system numbers at the head of the AS path against a sequence of one or more values or parameters, use the `as-path neighbor-is` command in route-policy configuration mode.

```
    as-path neighbor-is as-number-list [exact]
```

**Syntax Description**

- **as-number-list**: Numbers or parameters, enclosed in single quotation marks, that represent a sequence of autonomous system numbers.
  - Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.
  - Range for 4-byte Autonomous system numbers (ASNs) in asplain format is 1 to 4294967295.
  - Range for 4-byte Autonomous system numbers (ASNs) is asdot format is 1.0 to 65535.65535.

- **exact**: (Optional) Specifies that with the `exact` keyword, the `as-number-list` value must identically match the AS path for the route; without the `exact` keyword, any element in the `as-number-list` argument matches one or more occurrences of that element in the AS path for the route.

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

- **Release 6.0**: This command was introduced.

**Usage Guidelines**

Use the `as-path neighbor-is` command as a conditional expression within an `if` statement to test the autonomous system number or numbers at the head of the AS path against a sequence of one or more integral values or parameters. In other words, to test to learn if the sequence of autonomous system numbers matches the path beginning with the neighboring autonomous system from which this route was heard.

**Note**

For a list of all conditional expressions available within an `if` statement, see the `if` command.

This command has an equivalent regular expression (ios-regex). For example, AS path neighbor-is ‘1’ would be ‘^1’.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following are incomplete configuration examples:
These statements evaluate to true when the first autonomous system numbers on the AS path match, in the same order, the supplied parameters or integer values in the `neighbor-is` statement. If the neighboring autonomous system location happens to be an AS-set, the operator evaluates to true if the corresponding argument to the `neighbor-is` operator is an element of the AS-set.

Without the `exact` keyword, repeated autonomous system numbers in the AS path are ignored. For example,

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path neighbor-is '10 20' then
```

matches an AS path beginning

```
10 10 10 20 ...
```

and an AS path beginning:

```
10 20 ....
```

With the `exact` keyword, repetitions are not ignored, therefore

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path neighbor-is '10 20' exact then
```

matches the second of these AS paths but not the first.
as-path originates-from

To compare an AS path against the AS sequence beginning with the AS number that originated a route, use the **as-path originates-from** command in route-policy configuration mode.

**as-path originates-from as-number-list [exact]**

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>as-number-list</th>
<th>Numbers or parameters, enclosed in single quotation marks, that represent a sequence of autonomous system numbers.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>• Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Range for 4-byte Autonomous system numbers (ASNs) in asplain format is 1 to 4294967295.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Range for 4-byte Autonomous system numbers (ASNs) is asdot format is 1.0 to 65535.65535.</td>
</tr>
<tr>
<td>exact</td>
<td>(Optional) Specifies that with the exact keyword, the as-number-list value must identically match the AS path for the route; without the exact keyword, any element in the as-number-list argument matches one or more occurrences of that element in the AS path for the route.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command Default</th>
<th>No default behavior or values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Modes</td>
<td>Route-policy configuration</td>
</tr>
<tr>
<td>Command History</td>
<td><strong>Release</strong> Modification</td>
</tr>
<tr>
<td></td>
<td>Release 6.0 This command was introduced.</td>
</tr>
</tbody>
</table>

| Usage Guidelines  | Use the as-path originates-from command as a conditional expression within an if statement to compare an AS path to the autonomous system sequence. |

<table>
<thead>
<tr>
<th>Note</th>
<th>For a list of all conditional expressions available within an if statement, see the if command.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>The originates-from operator is similar to the neighbor-is operator, except that it looks at the autonomous system number at the opposite end of the AS path. In other words, it is comparing to the autonomous system that originated the route. It can take numbers or parameters, enclosed in single quotation marks, that represent a sequence of autonomous system numbers. When more than one number is specified in the list, the sequence of autonomous system numbers listed must appear as a subsequence in the AS path, with the last number corresponding to the autonomous system that originated the route.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>route-policy read, write</td>
</tr>
</tbody>
</table>
The following are incomplete configuration examples:

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path originates-from '10 11' then
RP/0/RP0/CPU0:router(config-rpl-if)# if as-path originates-from '$asnum 11' then
```

The first line of the preceding example evaluates to true if autonomous system 11 originated the route and then advertised it to autonomous system 10, from which the route was eventually propagated to us. In the case where the route has been aggregated, and the location of the originating autonomous system contains an AS-set, the `originates-from` operator evaluates to true if the argument to the `originates-from` operator is contained in the AS-set.

Without the `exact` keyword, repeated autonomous system numbers in the AS path are ignored. For example,

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path originates-from '10 11' then
```

matches an autonomous system path ending

```
...10 10 10 11
```

and an autonomous system path ending

```
...10 11
```

With the `exact` keyword, repetitions are not ignored, therefore

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path originates-from '10 11' exact then
```

matches the second of these autonomous system paths but not the first.
as-path passes-through

To verify if the supplied integer or parameter appears anywhere in the AS path or if the supplied sequence of integers and parameters appears, in the same order, anywhere in the AS path, use the `as-path passes-through` command in route-policy configuration mode.

```
as-path passes-through as-number-list [exact]
```

**Syntax Description**

- `as-number-list` Numbers or parameters, enclosed in single quotation marks, that represent a sequence of autonomous system numbers.
  - Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.
  - Range for 4-byte Autonomous system numbers (ASNs) in asplain format is 1 to 4294967295.
  - Range for 4-byte Autonomous system numbers (ASNs) as adot format is 1.0 to 65535.65535.

- `exact` (Optional) Specifies that with the `exact` keyword, the `as-number-list` value must identically match the AS path for the route; without the `exact` keyword, any element in the `as-number-list` argument matches one or more occurrences of that element in the AS path for the route.

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `as-path passes-through` command as a conditional expression within an `if` statement to verify if the specified integer or parameter appears anywhere in the AS path or if the sequence of integers and parameters appears.

**Note**

For a list of all conditional expressions available within an `if` statement, see the `if` command.

The `passes-through` operator takes a sequence of integers or parameters, enclosed in single quotation marks, as an argument. It can also take a single integer or parameter as an argument. It evaluates to true if the supplied integer or parameter appears anywhere in the AS path, or if the supplied sequence of integers and parameters appears, in the same order, anywhere in the AS path. This includes the `originates-from` or `neighbor-is` location in the AS path.
The following are incomplete configuration examples:

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path passes-through '10' then
RP/0/RP0/CPU0:router(config-rpl-if)# if as-path passes-through '$asnum' then
RP/0/RP0/CPU0:router(config-rpl-if)# if as-path passes-through '10 11' then
RP/0/RP0/CPU0:router(config-rpl-if)# if as-path passes-through '10 $asnum 12' then
```

Without the **exact** keyword, repeated autonomous system numbers in the AS path are ignored. For example:

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path passes-through '9 10 11' then
```

matches an AS path containing

```
...9 10 10 10 11....
```

and an AS path containing:

```
...9 10 11...
```

With the **exact** keyword, repetitions are not ignored. Therefore:

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path passes-through '9 10 11' exact then
```

matches the second of these AS paths but not the first.
as-path-set

To create a named AS path set, use the `as-path-set` command in XR Config mode. To remove the named AS path set, use the `no` form of this command.

```
as-path-set name
no as-path-set name
```

**Syntax Description**

- `name` Name of the AS path set.

**Command Default**

No default behavior or values

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `as-path-set` command to create a named AS path set.

An AS path set comprises operations for matching an AS path attribute.

This command enters AS path set configuration mode, in which you can use any of the below option to specify an operation.

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dfa-regex</td>
<td>Indicates the DFA (deterministic finite automata) style regular expression. It performs better for complex regular expressions. Single quotation marks are required around the regular expression.</td>
</tr>
<tr>
<td>ios-regex</td>
<td>Indicates the traditional IOS style regular expression. It performs better with simpler regular expressions. Single quotation marks are required around the regular expression.</td>
</tr>
<tr>
<td>length</td>
<td>Indicates the number of ASN (Autonomous System Number) in the AS path of a Border Gateway Protocol (BGP) route.</td>
</tr>
<tr>
<td>neighbor-is</td>
<td>Indicates the neighbor's AS-path number that can be matched with.</td>
</tr>
<tr>
<td>originates-from</td>
<td>Indicates the BGP AS from which the route originated.</td>
</tr>
<tr>
<td>passes-through</td>
<td>Indicates if the supplied integer or parameter appears anywhere in the AS path, or if the supplied sequence of integers and parameters appear, in the same order, anywhere in the AS path.</td>
</tr>
</tbody>
</table>
The above options can also be used as an inline set in a parenthesized list of comma-separated expressions.

### Options

<table>
<thead>
<tr>
<th>Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>unique-length</td>
<td>Indicates the length of BGP AS-path, ignoring duplicates.</td>
</tr>
</tbody>
</table>

### Examples

The following is a sample definition of an AS path set named aset1. This AS path set is composed of two elements. When used in a matching operation, this AS path set matches any route whose AS path ends with either the autonomous system number 42 or 127.

```
RP/0/RP0/CPU0:router(config)# as-path-set aset1
RP/0/RP0/CPU0:router(config-as)# ios-regex '_42$'
RP/0/RP0/CPU0:router(config-as)# ios-regex '_127$'
RP/0/RP0/CPU0:router(config-as)# end-set
```

The following is a sample of the as-path options used as an inline set.

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path in (ios-regex '_42$', ios-regex '_127$')
RP/0/RP0/CPU0:router(config-rpl-if)# pass
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
as-path unique-length

To perform specific checks based on the length of the AS path (match against the number of unique ASNs in the AS path), use the **as-path unique-length** command in route-policy configuration mode.

**as-path unique-length** \{eq|is|ge|le\} \{numberparameter\}

**Syntax Description**
- **eq | is | ge | le** Equal to; greater than or equal to; less than or equal to.
- **number** Value assigned to an 11-bit unsigned integer. Range is from 0 to 2047.
- **parameter** Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the **as-path unique-length** command as a conditional expression within an **if** statement to perform a match based on the length of the AS path.

**Note**
For a list of all conditional expressions available within an **if** statement, see the **if** command.

The **unique-length** operator is similar to the length operator, except that when an AS path has been padded with the same autonomous system number multiple times, the operator counts only one when the route is padded. Therefore, given an AS path of 333 333 111 222 123 444 444 444, the **unique-length** operator would return a value of 5, whereas the length operator would return a value of 8.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following examples show how to perform checks based on the AS path length. If the AS path matches the specified values, the local preference is set to 100:

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path unique-length eq 10 then
RP/0/RP0/CPU0:router(config-rpl-if)# if as-path unique-length ge 10 then
RP/0/RP0/CPU0:router(config-rpl-if)# if as-path unique-length le 10 then
RP/0/RP0/CPU0:router(config-rpl)# if as-path unique-length eq $integerparam then
RP/0/RP0/CPU0:router(config-rpl-if)# if as-path unique-length ge $geparam then
RP/0/RP0/CPU0:router(config-rpl-if)# if as-path unique-length le $leparam then
```
RP/0/RP0/CPU0:router(config-rpl)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl)# endif
community is-empty

To check if a route has no community attributes associated with it, use the **community is-empty** command in route-policy configuration mode.

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**

Use the **community is-empty** command as a conditional expression within an **if** statement to check if a route has community attributes associated with it.

**Note**

For a list of all conditional expressions available within an **if** statement, see the **if** command.

This command takes no arguments and evaluates to true only if the route has no community attributes associated with it.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, if the route has no community attributes associated with it, then the local preference is set to 100:

```
RP/0/RP0/CPU0:router(config-rpl)# if community is-empty then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# endif
```
community matches-any

To match any elements of a community set, use the **community matches-any** command in route-policy configuration mode.

**community matches-any** {community-set-name|inline-community-set|parameter}

**Syntax Description**
- community-set-name: Name of a community set.
- inline-community-set: Inline community set. The inline community set must be enclosed in parentheses.
- parameter: Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**
- Release 6.0: This command was introduced.

**Usage Guidelines**
Use the **community matches-any** command as a conditional expression within an `if` statement to match any element of a community set.

For a list of all conditional expressions available within an `if` statement, see the `if` command.

**Note**
A simple condition using the **matches-any** operator evaluates as true if at least one community element of the community attribute for the route matches an element in the community set operand. If no community in the route matches any of the specifications in the named or inline set, then the condition evaluates to false. Likewise, when there is no community at all in the route, the condition evaluates to false.

Matching of a community in the route to a specification in a named or an inline set is intuitive. If the community specification in a set is the familiar colon-separated decimal 16-bit numbers specification, or one of the well-known communities, the community matches the specification if the specification denotes the same 32-bit number as that in the route. If the community specification uses a wildcard, then the community in the route matches if it is one of the many communities denoted by the wildcard specification. In inline sets, community specifications may be parameterized, in which case the relevant matching is done when the value of the parameter has been supplied.

Communities may also be matched using range and regular expression operators. Range specifications are entered as follows: `[low-value .. high-value]`. Either or both colon-separated halves of a community value may contain a range. The following are valid range specifications:

```
10:[100..1000]
[10..100]:80
[10..100]:[100..2000]
```
In addition, the `private-as` keyword may be used to specify the range from 64512 to 65534. Regular expressions are specified as the `ios-regex` keyword followed by a valid regular expression string.

Community values from the route are matched one at a time to the match specifications. Therefore, regex match specifications are expected to represent one individual community value and not a sequence of community values.

### Task ID Operations

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

In the following example, a named community set called `my-community-set` and a route policy called `community-matches-any-example` are created. The policy sets the local-preference to 100 for any route that has one or more of the communities in the `my-community-set` community set. If the route does not have any of these communities, the policy checks whether it has any communities whose first half is in the range from 10 to 25 and whose second half is the value 35, in which case it sets the local-preference to 200. Otherwise, it checks for a community value in the range of 30:100 to 30:500, in which case it sets the local-preference to 300.

```
RP/0/RP0/CPU0:router(config)# community-set my-community-set
RP/0/RP0/CPU0:router(config-comm)# 10:20,
RP/0/RP0/CPU0:router(config-comm)# 10:30,
RP/0/RP0/CPU0:router(config-comm)# 10:40
RP/0/RP0/CPU0:router(config-comm)# end-set

RP/0/RP0/CPU0:router(config)# route-policy community-matches-any-example
RP/0/RP0/CPU0:router(config-rpl)# if community matches-any my-community-set then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# elseif community matches-any ([10..25]:35) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 200
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif community matches-any (30:[100..500]) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 300
RP/0/RP0/CPU0:router(config-rpl-elseif)# endif
RP/0/RP0/CPU0:router(config-rpl)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
community matches-every

To match every element of a community set, use the **community matches-every** command in route-policy configuration mode.

```
community matches-every \{community-set-name inline-community-set parameter\}
```

**Syntax Description**

- **community-set-name**  Name of a community set.
- **inline-community-set**  Inline community set. The inline community set must be enclosed in parentheses.
- **parameter**  Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

**Release**  **Modification**

Release 6.0  This command was introduced.

**Usage Guidelines**

Use the **community matches-every** command as a conditional expression within an **if** statement to match every element of a community set.

For a list of all conditional expressions available within an **if** statement, see the **if** command.

A simple condition using the **matches-every** operator evaluates as true if every specification in the named set or inline set specified matches at least one community value in the route. If any community specification in the named or inline set is not matched, then the operation evaluates to false.

Matching of a community in the route to a specification in a named or an inline set is intuitive. If the community-specification in a set is the familiar colon-separated decimal 16-bit numbers specification, or one of the well-known communities, the community matches the specification if the specification denotes the same 32-bit number as that in the route. If the community specification uses a wildcard, then the community in the route matches if it is one of the many communities denoted by the wildcard specification. In inline sets, community specifications may be parameterized, in which case the relevant matching is done when the value of the parameter has been supplied.

Communities may also be matched using range and regular expression operators. Range specifications are entered as follows: `[low-value .. high-value]`. Either or both colon-separated halves of a community value may contain a range. The following are valid range specifications:

- `10:[100..1000]`
- `[10..100]:80`
- `[10..100]:[100..2000]`
Therefore, a **matches-every** operation with two community range specifications means that a community must be present in the route that corresponds to each range. For example, in the following statement:

```
if community matches-every {10:[100..200],20:[100..200]) then
```

the statement evaluates as true if one or more communities in the route lie in the range 10:[100.200] and one or more communities in the route lie in the range 20:[100..200].

In addition, the **private-as** keyword may be used to specify the range from 64512 to 65534.

Regular expressions are specified as the **ios-regex** keyword followed by a valid single-quoted regular expression string. Community values from the route are matched one at a time against the match specifications. Therefore, regex match specifications are expected to represent one individual community value and not a sequence of community values.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

In the following example, the route policy named community-matches-every-example sets the local-preference value to 100 for all routes that have all three communities in the my-community-set community set. Routes that do not have all three communities but have a community that matches the first regular expression match have the local-preference value set to 200. Finally, any remaining routes that match the last regular expression have the local-preference values set to 300.

```
RP/0/RP0/CPU0:router(config)# community-set my-community-set
RP/0/RP0/CPU0:router(config-comm)# 10:20,
RP/0/RP0/CPU0:router(config-comm)# 10:30,
RP/0/RP0/CPU0:router(config-comm)# 10:40
RP/0/RP0/CPU0:router(config-comm)# end-set

RP/0/RP0/CPU0:router(config)# route-policy community-matches-every-example
RP/0/RP0/CPU0:router(config-rpl)# if community matches-every my-community-set then
RP/0/RP0/CPU0:router(config-rp-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rp-elseif)# elseif community matches-every (ios-regex '
10:[0-9]0_' ) then
RP/0/RP0/CPU0:router(config-rp-elseif)# set local-preference 200
RP/0/RP0/CPU0:router(config-rp-elseif)# elseif community matches-every (ios-regex '
10:[0-9]0_' ) then
RP/0/RP0/CPU0:router(config-rp-elseif)# set local-preference 300
RP/0/RP0/CPU0:router(config-rp-elseif)# endif
RP/0/RP0/CPU0:router(config-rp)# end-policy
```
community-set

To define a community set, use the `community-set` command in XR Config mode. To remove the community set, use the `no` form of this command.

```
community-set name
no community-set name
```

**Syntax Description**

- `name` Name of the community set.

**Command Default**

No default behavior or values

**Command Modes**

XR Config mode

**Command History**

- **Release** 6.0  This command was introduced.

**Usage Guidelines**

Regular expressions and ranges can be specified to match the communities. An attempt to use a community set that contains a range or regular expression to set a community value is rejected when an attempt to attach such a policy is made.

A community set holds community values for matching against the Border Gateway Protocol (BGP) community attribute. A community is a 32-bit quantity. For notational convenience, each community value must be split in half and expressed as two unsigned decimal integers in the range from 0 to 65535, separated by a colon.

The inline form of a community set also supports parameterization. Each 16-bit portion of the community may be parameterized.

The routing policy language (RPL) provides symbolic names for the standard well-known community values: `accept-own` is 0xFFFF0001, `internet` is 0:0, `no-export` is 65535:65281, `no-advertise` is 65535:65282, and `local-as` is 65535:65283.

RPL also provides a facility for using wildcards in community specifications. A wildcard is specified by inserting an asterisk (*) in place of one of the 16-bit portions of the community specification, which indicates that any value for that portion of the community matches.

Every community set must contain at least one community value. An empty community set is invalid and the policy configuration system rejects it.

Community sets can be entered in these formats:

<table>
<thead>
<tr>
<th>Format</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>#-remark</code></td>
<td>Remark beginning with '#!'</td>
</tr>
<tr>
<td>*</td>
<td>Wildcard (any community or part thereof)</td>
</tr>
<tr>
<td><code>0-65535</code></td>
<td>16-bit half-community number</td>
</tr>
<tr>
<td>`</td>
<td>`</td>
</tr>
<tr>
<td><code>accept-own</code></td>
<td>Accept-Own (BGP well-known community)</td>
</tr>
<tr>
<td>Format</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>dfa-regex</td>
<td>DFA (deterministic finite automata) style regular expression</td>
</tr>
<tr>
<td>internet</td>
<td>Internet (BGP well-known community)</td>
</tr>
<tr>
<td>ios-regex</td>
<td>Traditional IOS style regular expression</td>
</tr>
<tr>
<td>local-AS</td>
<td>Do not send outside local AS (BGP well-known community)</td>
</tr>
<tr>
<td>no-advertise</td>
<td>Do not advertise to any peer (BGP well-known community)</td>
</tr>
<tr>
<td>no-export</td>
<td>Do not export to next AS (BGP well-known community)</td>
</tr>
<tr>
<td>private-as</td>
<td>Match within BGP private AS range [64512..65534]</td>
</tr>
</tbody>
</table>

The dfa-regex and ios-regex syntax for community set is "["[^[^':>&< ];^[^':>&< ]^[^']". This means that regex starts with a single-quote ("') followed by a string of any character (that does not include single-quote, colon, ampersand, less-than, greater-than, or space) followed by a colon, and a string of any characters (that does not include single-quote, colon, ampersand, less-than, greater-than, or space) followed by single-quote.

### Examples

In the following example, a community set named cset_accept_own is created:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#community-set cset_accept_own
RP/0/RP0/CPU0:router(config-comm)#accept-own
RP/0/RP0/CPU0:router(config-comm)#end-set
```

In the following example, a community set named cset1 is created:

```
RP/0/RP0/CPU0:router(config)# community-set cset1
RP/0/RP0/CPU0:router(config-comm)# 12:34,
RP/0/RP0/CPU0:router(config-comm)# 12:56,
RP/0/RP0/CPU0:router(config-comm)# 12:78,
RP/0/RP0/CPU0:router(config-comm)# internet
RP/0/RP0/CPU0:router(config-comm)# end-set
```

In the following example, a community set named cset2 is created:

```
RP/0/RP0/CPU0:router(config)# community-set cset2
RP/0/RP0/CPU0:router(config-comm)# 123:456,
RP/0/RP0/CPU0:router(config-comm)# no-advertise,
```
In the following example, a community set named cset3 is created. This policy uses wildcards and matches all communities where the autonomous system part of the community is 123.

```plaintext
RP/0/RP0/CPU0:router(config-comm)# end-set

RP/0/RP0/CPU0:router(config)# community-set cset3
RP/0/RP0/CPU0:router(config-comm)# 123:*
RP/0/RP0/CPU0:router(config-comm)# end-set
```
delete community

To delete community attributes associated with a Border Gateway Protocol (BGP) route, use the `delete community` command in route-policy configuration mode.

```
delete community { all | in { community-set-name | inline-community-set | not in { community-set-name | inline-community-set } } }
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>all</code></td>
<td>Removes all communities except the well-known communities.</td>
</tr>
<tr>
<td><code>in</code></td>
<td>Removes any communities associated with the route that are listed in either the named community set or the inline community set.</td>
</tr>
<tr>
<td><code>community-set-name</code></td>
<td>Name of a community set.</td>
</tr>
<tr>
<td><code>inline-community-set</code></td>
<td>Inline community set. The inline community set must be enclosed in parentheses.</td>
</tr>
<tr>
<td><code>parameter</code></td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
<tr>
<td><code>not in</code></td>
<td>Removes all communities that are not listed in either the named community set or the inline community set, and are not well-known communities.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `delete community` command to delete community attributes associated with a BGP route.

**Note**

The `delete community` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

Communities are 32-bit values carried in Border Gateway Protocol (BGP) routes. Each route may have zero or more communities in an unordered list.

You can remove a well-known community (internet, no-export, no-advertise, or local-as) from a route, but this removal must be done explicitly. This command should be used with a degree of caution. In general, few circumstances exist in which you would need to remove a well-known community.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to delete any communities associated with the routes that are listed in either the named community set or inline community set, respectively.

```
RP/0/RP0/CPU0:router(config-rpl)# delete community in my_community_set
RP/0/RP0/CPU0:router(config-rpl)# delete community in (10:[0..50],20:[60..80])
```

The following example shows how to remove all communities including well-known communities.

```
RP/0/RP0/CPU0:router(config-rpl)# delete community in (internet, no-export, no-advertise, local-as, *:*)
```

The following example shows how to remove all communities except for the well-known communities.

```
RP/0/RP0/CPU0:router(config-rpl)# delete community all
```

The following example shows how to delete the well-known community value internet from a route:

```
RP/0/RP0/CPU0:router(config-rpl)# delete community in (internet)
```
**delete extcommunity rt**

To delete route target (RT) extended community attributes associated with a Border Gateway Protocol (route), use the `delete extcommunity rt` command in route-policy configuration mode.

```
delete extcommunity rt {all in {extcommunity-set-name inline-extcommunity-set parameter} not in {extcommunity-set-name inline-extcommunity-set parameter}}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>all</td>
<td>Removes all extended communities.</td>
</tr>
<tr>
<td>in</td>
<td>Removes any extended communities associated with the routes that are listed in either the named extended community set or the inline extended community set.</td>
</tr>
<tr>
<td>extcommunity-set-name</td>
<td>Name of an extended community set.</td>
</tr>
<tr>
<td>inline-extcommunity-set</td>
<td>Inline extended community set. The inline extended community set must be enclosed in parentheses.</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
<tr>
<td>not in</td>
<td>Removes all extended communities that are not listed in either the named extended community set or the inline extended community set, and are not well-known extended communities.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `delete extcommunity rt` command to delete extended community values from a BGP route target extended community list in a route.

**Note**

The `delete extcommunity rt` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

Extended communities are similar to regular Border Gateway Protocol (BGP) communities but contain more data and have a richer structure for encoding information in them.


Wildcards (*) and regular expressions are allowed for extended community set elements.

The forms of this command that take a named extended community set or an inline extended community set value as arguments are equivalent. They delete any extended communities that are listed in either the named set or the inline set, respectively.
### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

In the following example, all extended communities are deleted:

```
RP/0/RP0/CPU0:router(config-rpl)# delete extcommunity rt all
```

In this example, any extended communities that are listed in my-extcommunity-set are deleted:

```
RP/0/RP0/CPU0:router(config-rpl)# delete extcommunity rt in my-extcommunity-set
```

In this example, extended communities associated with the route listed in the named inline extended community sets are deleted:

```
RP/0/RP0/CPU0:router(config-rpl)# delete extcommunity rt in (67:29, 67:55)
```
destination in

To match a destination entry in a named prefix set or inline prefix set, use the `destination in` command in route-policy configuration mode.

```plaintext
destination in {prefix-set-name|inline-prefix-set|parameter}
```

**Syntax Description**

- `prefix-set-name` Name of a prefix set.
- `inline-prefix-set` Inline prefix set. The inline prefix set must be enclosed in parentheses.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `destination in` command as a conditional expression within an `if` statement to match a destination entry in a named prefix set or inline prefix set.

**Note**

For a list of all conditional expressions available within an `if` statement, see the `if` command.

This command takes either a named prefix set or an inline prefix set value as an argument. The condition returns true if the destination entry matches any entry in the prefix set or inline prefix set. An attempt to match a destination using a prefix set that is defined but contains no elements returns false.

The routing policy language (RPL) provides the ability to test destinations for a match to a list of prefix match specifications using the `in` operator. The `destination in` command is protocol-independent.

In Border Gateway Protocol (BGP), the destination of a route is also known as its network-layer reachability information (NLRI). It comprises a prefix value and a mask length.

RPL supports both 32-bit IPv4 prefixes, specified in dotted-decimal format, and 128-bit IPv6 prefixes, specified in colon-separated hexadecimal format.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, a prefix set named my-prefix-set is defined and a route policy named use-destination-in is created. Within the use-destination-in route policy, the `destination in` command
is used within an `if` statement to learn if the destination is in the prefix-set named `my-prefix-set`. If it is, then local preference is set to 100. If it is not in `my-prefix-set` but does match the next prefix specifications, then local preference is set to 200.

```plaintext
RP/0/RP0/CPU0:router(config)# prefix-set my-prefix-set
RP/0/RP0/CPU0:router(config-pfx)# 10.0.0.1/32,
RP/0/RP0/CPU0:router(config-pfx)# fe80::203:0:0:0/64,
RP/0/RP0/CPU0:router(config-pfx)# 10.0.0.2/24 le 32
RP/0/RP0/CPU0:router(config-pfx)# end-set

RP/0/RP0/CPU0:router(config)# route-policy use-destination-in
RP/0/RP0/CPU0:router(config-rpl)# if destination in my-prefix-set then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# elseif destination in (10.0.0.1/32, 10.0.0.2/24 le 32) then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 200
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```

In the following example, a prefix set named `ipv6-prefix-set` is defined and a route policy named `ipv6-destination-in` is created. Within the `ipv6-destination-in` route policy, the `destination in` command is used within an `if` statement to learn if the destination is in the prefix-set named `ipv6-prefix-set`. If it is, then the next-hop is set to 2001:abcd:fedc::1. If it is not in `ipv6-prefix-set` but does match the next prefix specifications, then the next-hop is set to 1111:2222:3333:4444:5555:6666:7777:8888.

```plaintext
RP/0/RP0/CPU0:router(config)# prefix-set ipv6-prefix-set
RP/0/RP0/CPU0:router(config-pfx)# 2001::1::1/64,
RP/0/RP0/CPU0:router(config-pfx)# 2001::2::1/64,
RP/0/RP0/CPU0:router(config-pfx)# 2001::3::1/64,
RP/0/RP0/CPU0:router(config-pfx)# 2001::4::1/64
RP/0/RP0/CPU0:router(config-pfx)# end-set

RP/0/RP0/CPU0:router(config)# route-policy ipv6-destination-in
RP/0/RP0/CPU0:router(config-rpl)# if destination in ipv6-prefix-set then
RP/0/RP0/CPU0:router(config-rpl-if)# set next-hop 2001:abcd:fedc::1
RP/0/RP0/CPU0:router(config-rpl-if)# elseif destination in (2001::1, 2002::1:3::/64) then
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**done**

To stop executing a policy and accept the route, use the `done` command in route-policy configuration mode.

### Syntax Description

This command has no arguments or keywords.

### Command Default

No default behavior or values

### Command Modes

Route-policy configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `done` command to stop executing the policy and accept the route.

**Note**

The `done` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

When encountering a `done` statement the route is passed and no further policy statements are executed. All modifications made to the route prior to the `done` statement are still valid.

**Note**

The default action of a route policy is to drop or discard any routes that have not been either explicitly passed or for which no attempt has been made to modify with an action. The routing policy language (RPL) does not have specific "match clauses," which means the default drop behavior is controlled by whether a route has been explicitly passed or an attempt has been to modify the route using an action statement.

### Examples

In the following example, if the destination match succeeds for 29.0.0.0/8 le 32, the execution continues past set community 102:12 and onto the next statement. If the destination match succeeds for 39.0.0.0/8 le 32 execution, then the policy execution stops when in encounters the `done` statement.

```
RP/0/RP0/CPU0:router(config)# route-policy done_st_example
RP/0/RP0/CPU0:router(config-rpl)# if destination in (29.0.0.0/8 le 32) then
RP/0/RP0/CPU0:router(config-rpl-if)# set community 102:12
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# if destination in (39.0.0.0/8 le 32) then
RP/0/RP0/CPU0:router(config-rpl-if)# set community 102:39
RP/0/RP0/CPU0:router(config-rpl-if)# done
```
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# if destination in (49.0.0.0/8 le 32) then
RP/0/RP0/CPU0:router(config-rpl-if)# set community 102:49
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# if destination in (59.0.0.0/8 le 32) then
 RP/0/RP0/CPU0:router(config-rpl-if)# set community 102:59
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
drop

To discard a route, use the `drop` command in route-policy configuration mode.

**Syntax Description**
This command has no arguments or keywords.

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `drop` command within a route policy to drop a route.

**Note**
The `drop` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

This command causes the route to be dropped. After a route is dropped, no further execution of policy occurs. Therefore, if after executing the first two statements of a policy the `drop` statement is encountered, the route is discarded and execution stops immediately even when the policy contains further statements.

**Note**
The default action of a route policy is to drop or discard any routes that have not been either explicitly passed or attempted to be modified with an action. The routing policy language (RPL) does not have specific “match clauses,” which means the default drop behavior is controlled by whether a route has been explicitly passed or an attempt has been to modify the route using an action statement.

**Examples**
In the following example, any route with a destination address contained within the prefix set pset1 is dropped:

```
RP/0/RP0/CPU0:router(config-rpl)# if destination in pset1 then
RP/0/RP0/CPU0:router(config-rpl-if)# drop
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
## Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>route-policy</code></td>
<td>Edits the contents of a route policy.</td>
</tr>
<tr>
<td><code>prefix-set</code></td>
<td>Edits the contents of a prefix set.</td>
</tr>
<tr>
<td><code>as-path-set</code></td>
<td>Edits the contents of an AS path set.</td>
</tr>
<tr>
<td><code>community-set</code></td>
<td>Edits the contents of a community set.</td>
</tr>
<tr>
<td><code>extcommunity-set</code></td>
<td>Edits the contents of an extended community set of the specified type.</td>
</tr>
<tr>
<td><code>rt</code></td>
<td>Edits the BGP route target (RT) extended community.</td>
</tr>
<tr>
<td><code>soo</code></td>
<td>Edits the BGP site of origin (SoS) extended community.</td>
</tr>
<tr>
<td><code>policy-global</code></td>
<td>Edits the contents of policy-global definitions.</td>
</tr>
<tr>
<td><code>rd-set</code></td>
<td>Edits the contents of a route-distinguisher set.</td>
</tr>
<tr>
<td><code>name</code></td>
<td>Name of a route policy, a prefix set, an AS path set, a community set, or an extended community set, RD set, or global parameters.</td>
</tr>
<tr>
<td><code>nano</code></td>
<td>(Optional) Uses GNU Nano text editor.</td>
</tr>
<tr>
<td><code>emacs</code></td>
<td>(Optional) Uses Micro Emacs editor.</td>
</tr>
<tr>
<td><code>vim</code></td>
<td>(Optional) Uses VI Improved editor.</td>
</tr>
<tr>
<td><code>inline</code></td>
<td>(Optional) Uses the command line.</td>
</tr>
<tr>
<td><code>add</code></td>
<td>Appends the element to the set.</td>
</tr>
<tr>
<td><code>prepend</code></td>
<td>Prepends the element to the set.</td>
</tr>
<tr>
<td><code>remove</code></td>
<td>Removes the element from the set.</td>
</tr>
<tr>
<td><code>set-element</code></td>
<td>Value of the set element.</td>
</tr>
</tbody>
</table>

### Note
To inline edit multiple set elements separated with comma, use quotes to club the entries as a single argument. Example:
```
edit extcommunity-set rt rt_set inline add "4:4,5:4"
```
**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `edit` command to edit the contents of a route policy, a prefix set, an AS path set, a community set, an extended community set, a global policy, or a route destination set.

After editing with Nano, save the edit buffer and exit the editor using the Ctrl-X keystroke.

After editing with Emacs, save the editor buffer by using the Ctrl-X and Ctrl-S keystrokes. To save and exit the editor, use the Ctrl-X and Ctrl-C keystrokes.

After editing with VIM, to write to a current file and exit use the :wq or :x or ZZ keystrokes. To quit and confirm, use the :q keystrokes. To quit and discard changes, use the :q! keystrokes.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, the policy_A policy is opened in the editor:

```
RP/0/RP0/CPU0:router# edit route-policy policy_A
```

```diff
---
  if destination in (2001::/8) then
drop
endif
end-policy!
```

```
RP/0/RP0/CPU0:router# edit route-policy policy_B
```

```diff
---
  set metric-type type_1
  if destination in (2001::/8) then
drop
endif
end-policy!
```

If there are parse errors, you are asked whether editing should continue:

```
% Syntax/Authorization errors in one or more commands.!! CONFIGURATION
```
FAILED DUE TO SYNTAX/AUTHORIZATION ERRORS
set metric-type type_1
if destination in (2001::/8) then
drop
endif
end-policy
!

Continue editing? [no]:

If you answer yes, the editor continues on the text buffer from where you left off. If you answer no, the running configuration is not changed and the editing session is ended.

After the policy is opened, it may be manipulated using normal editor commands, then saved and committed to the running configuration.
end-global

To end the definition of global parameters and exit global parameter configuration mode, use the **end-global** command in global parameter configuration mode.

### Syntax Description

This command has no arguments or keywords.

### Command Default

No default behavior or values

### Command Modes

Global parameter configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the **end-global** command to end the definition of global parameters and exit global parameter configuration mode.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

In the following example, the **end-global** command ends the definition of global parameters:

```plaintext
RP/0/RP0/CPU0:router (config)# policy-global
RP/0/RP0/CPU0:router (config-rp-gl)# glbpathype 'ebgp'
RP/0/RP0/CPU0:router (config-rp-gl)# glbtag '100'
RP/0/RP0/CPU0:router (config-rp-gl)# end-global
```
end-policy

To end the definition of a route policy and exit route-policy configuration mode, use the `end-policy` command in route-policy configuration mode.

### Syntax Description
This command has no arguments or keywords.

### Command Default
No default behavior or values

### Command Modes
Route-policy configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines
Use the `end-policy` command to end the definition of a route policy and exit route-policy configuration mode.

### Examples
In the following example, the `end-policy` command ends the definition of a route policy:

```
RP/0/RP0/CPU0:router(config)#route-policy med-to-local-pref
RP/0/RP0/CPU0:router(config-rpl)#if med eq 150 then
  RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 10
  
elseif med eq 200 then
  RP/0/RP0/CPU0:router(config-elseif)# set local-preference 60
  
elseif med eq 250 then
  RP/0/RP0/CPU0:router(config-elseif)# set local-preference 0
  endif

RP/0/RP0/CPU0:router(config-elseif)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
end-set

To end the definition of an AS path set, a prefix set, a community set, an extended community set, or an RD set and return to XR Config mode, use the end-set command in route-policy configuration mode.

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

AS path set configuration
Prefix set configuration
Community set configuration
Extended community set configuration
Route distinguisher set configuration

Command History

Release   Modification
Release 6.0  This command was introduced.

Usage Guidelines

Use the end-set command to end the definition of an AS path set, a prefix set, a community set, or an extended community set.

Examples

In the following example, the end-set command ends the definition of an AS path set named aset1:

```
RP/0/RP0/CPU0:router(config)# as-path-set aset1
RP/0/RP0/CPU0:router(config-as)# ios-regex '_42$',
RP/0/RP0/CPU0:router(config-as)# ios-regex '_127$'
RP/0/RP0/CPU0:router(config-as)# end-set
RP/0/RP0/CPU0:router(config)#
```

The following example shows how to create an RD set called my_rd_set and use the end-set command to end the definition:

```
RP/0/RP0/CPU0:router(config)# rd-set my_rd_set
RP/0/RP0/CPU0:router(config-rd)# 172.16.0.0/16:*,
RP/0/RP0/CPU0:router(config-rd)# 172.17.0.0/16:100,
RP/0/RP0/CPU0:router(config-rd)# 192:*,
RP/0/RP0/CPU0:router(config-rd)# 192:100
```
RP/0/RP0/CPU0:router(config-rd)# **end-set**
extcommunity rt is-empty

To check if a Border Gateway Protocol (BGP) route has route target (RT) extended community attributes associated with it, use the `extcommunity rt is-empty` command in route-policy configuration mode.

**Syntax Description**
This command has no arguments or keywords.

**Command Default**
No default behavior or value

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `extcommunity rt is-empty` command as a conditional expression within an `if` statement to check if a BGP route has extended community attributes associated with it.

**Note**
For a list of all conditional expressions available within an `if` statement, see the `if` command.

The `is-empty` operator takes no arguments and evaluates to true if the route has no extended community attributes associated with it.

**Examples**

In the following example, if the extended community is empty, then the local preference is set to 100:

```
RP/0/RP0/CPU0:router(config)# route-policy extcommunity-is-empty-example
RP/0/RP0/CPU0:router(config-rpl)# if extcommunity rt is-empty then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100

RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
### extcommunity rt matches-any

To match any element of a Border Gateway Protocol (BGP) route target (RT) extended community set, use the `extcommunity rt matches-any` command in route-policy configuration mode.

```
extcommunity rt matches-any {extcommunity-set-name|inline-extcommunity-set|parameter}
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><code>extcommunity-set-name</code></td>
<td>Name of an RT extended community set.</td>
</tr>
<tr>
<td><code>inline-extcommunity-set</code></td>
<td>Inline RT extended community set. The inline extended community set must be enclosed in parentheses.</td>
</tr>
<tr>
<td><code>parameter</code></td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
</tbody>
</table>

#### Command Default

No default behavior or values

#### Command Modes

Route-policy configuration

#### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

#### Usage Guidelines

Use the `extcommunity rt matches-any` command as a conditional expression within an `if` statement to match elements of an extended community set.

For a list of all conditional expressions available within an `if` statement, see the `if` command.

A simple condition using the `matches-any` operator evaluates as true if at least one extended community in the route matches an extended community specification in the named or inline set. If no extended community in the route matches any of the specifications in the named or inline set, then this simple condition evaluates to false. Likewise, when there is no extended community at all in the route, the condition evaluates to false.

Matching an extended community in the route to a specification in a named or an inline set is intuitive. In inline sets, extended community specifications may be parameterized, in which case the relevant matching is done when the value of the parameter has been supplied.

#### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

#### Examples

In the following example, an extended community set named `my-extcommunity-set` and a parameterized route-policy named `my-extcommunity-set-example($tag,$ip)` are defined. The `extcommunity rt matches-any` command is used in an if statement such that if at least one extended community in the route matches an extended community specification in the named set, then the local preference is set to 100. If there is no extended community in the route that matches any of the
specifications in the named set, then the condition evaluates as false and the extended community is compared to the inline extended sets.

```
RP/0/RP0/CPU0:router(config)# extcommunity-set rt my-extcommunity-set
RP/0/RP0/CPU0:router(config-ext)# 10:615,
RP/0/RP0/CPU0:router(config-ext)# 10:6150,
RP/0/RP0/CPU0:router(config-ext)# 15.15.15.15
RP/0/RP0/CPU0:router(config-ext)# end-set

RP/0/RP0/CPU0:router(config)# route-policy my-extcommunity-set-example($tag,$ip)
RP/0/RP0/CPU0:router(config-rpl)# if extcommunity rt matches-any my-extcommunity-set then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# elseif extcommunity rt matches-any (10:20, 10:$tag) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 200
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif extcommunity rt matches-any ($ip:$tag) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 300
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif extcommunity rt matches-any (2.3.4.5:$tag) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 400
RP/0/RP0/CPU0:router(config-rpl-elseif)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
extcommunity rt matches-every

To match every element of a Border Gateway Protocol (BGP) route target (RT) extended community set, use the `extcommunity rt matches-every` command in route-policy configuration mode.

```
extcommunity rt matches-every {extcommunity-set-name|inline-extcommunity-set|parameter}
```

**Syntax Description**
- `extcommunity-set-name` Name of an RT extended community set.
- `inline-extcommunity-set` Inline RT extended community set. The inline extended community set must be enclosed in parentheses.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Use the `extcommunity rt matches-every` command as a conditional expression within an `if` statement to match every element of an RT extended community set.

**Note**
For a list of all conditional expressions available within an `if` statement, see the `if` command.

A simple condition using the `matches-every` operator evaluates as true if every extended community value in the extended community attribute for the route matches at least one element of the extended community set or inline set. If no extended community in the route matches any of the specifications in the named or inline set, then this simple condition evaluates to false. Likewise, when there is no extended community at all in the route, the condition evaluates to false.

Matching an extended community in the route to a specification in a named or an inline set is intuitive. In inline sets, extended community specifications may be parameterized, in which case the relevant matching is done when the value of the parameter has been supplied.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, an extended community set named `my-extcommunity-set` and a parameterized route policy named `extcommunity-matches-every-example ($a, $tag)` are defined. The condition `extcommunity rt matches-every` is used in an `if` statement in this policy. If it evaluates to true, the local-preference value is set to 100. If it evaluates to false, the extended community is
evaluated using an inline set. If that condition evaluates to true, the local-preference value is set to 200. If it evaluates to false, the local-preference value is set to 300.

RP/0/RP0/CPU0:router(config)# extcommunity-set rt my-extcommunity-set
RP/0/RP0/CPU0:router(config-ext)# 10:20,
RP/0/RP0/CPU0:router(config-ext)# 10:30,
RP/0/RP0/CPU0:router(config-ext)# 10:40
RP/0/RP0/CPU0:router(config-ext)# end-set
RP/0/RP0/CPU0:router(config)# route-policy extcommunity-matches-every-example($as,$tag)
RP/0/RP0/CPU0:router(config-rpl)# if extcommunity rt matches-every my-extcommunity-set then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# elseif extcommunity rt matches-every (10:20, 10:$tag, $as:30) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 200
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 300
RP/0/RP0/CPU0:router(config-rpl-elseif)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
**extcommunity rt matches-within**

To match at least one element of an extended community set of a Border Gateway Protocol (BGP) route target (RT), use the `extcommunity rt matches-within` command in route-policy configuration mode.

```
extcommunity rt matches-within {rt-type-extcommunity-set-nameinline-extcommunity-setparameter}
```

**Syntax Description**

- `rt-type-extcommunity-set-name` Name of an RT extended community set.
- `inline-extcommunity-set` Inline RT extended community set, enclosed in parentheses.
- `parameter` Parameter name preceded with a “$” symbol.

**Command Default**

None

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `extcommunity rt matches-within` command as a conditional expression within an if statement to match elements of an extended community set.

For a list of all conditional expressions available within an if statement, see the `if` command.

A simple condition using the matches-within operator evaluates as true if all the elements in extended community from the route match any element in the extended community set. For example, let ‘c’ be the RTs from the route and ‘m’ be the RT set from the policy. With the `extcommunity rt matches-within` configuration, each value in ‘c’ must match any (or at least one) value in ‘m’.

Matching an extended community in the route to a specification in a named or an inline set is intuitive. In inline sets, extended community specifications may be parameterized, in which case the relevant matching is done when the value of the parameter has been supplied.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

In the following example, an extended community set named `my-extcommunity-set` and a parameterized route-policy named `my-extcommunity-set-example($tag,$ip)` are defined. The `extcommunity rt matches-within` command is used in an if statement such that if all extended community values in the route match any element of the extended community specification in the named set, then the local preference is set to 100.

```
RP/0/RP0/CPU0:router(config)#extcommunity-set rt my-extcommunity-set
```
route-policy my-extcommunity-set-example($tag,$ip)
if extcommunity rt matches-within my-extcommunity-set then
set local-preference 100
extcommunity-set cost

To define a cost extended community set, use the `extcommunity-set cost` command in XR Config mode. To remove the cost extended community set, use the `no` form of this command.

```
extcommunity-set cost name
no extcommunity-set cost name
```

**Syntax Description**

- **name**  
  Name of a cost extended community set. The `name` argument is case sensitive, can contain any alphanumeric characters, and can be up to 63 characters in length.

**Command Default**

No default behavior or values

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**

Use the `extcommunity-set cost` command to define a cost extended community set.

An extended community set is analogous to a community set except that it contains extended community values instead of regular community values. Extended community values are 64-bit structured values. An extended community set also supports named forms and inline forms.

Cost extended communities can be entered in these formats:

- `#-remark` --- Remark beginning with '#'
- `0-255` --- Decimal number
- `abort` --- Discard RPL definition and return to top level config
- `end-set` --- End of set definition
- `exit` --- Exit from the submode
- `igp:` --- Cost Community with IGP as point of insertion
- `pre-bestpath:` --- Cost Community with Pre-Bestpath as point of insertion
- `show` --- Show partial RPL configuration

Multiple cost community set clauses can be configured in each route policy block or sequence. Each cost community set clause must have a different ID (0-255). The cost community set clause with the lowest cost-value is preferred by the best path selection process when all other attributes are equal.

As with community sets, the inline form supports parameterization within parameterized policies. Either portion of the extended community value can be parameterized.

Every extended community set must contain at least one extended community value. Empty extended community sets are invalid and the policy configuration system rejects them.

Wildcards (*) and regular expressions are allowed for extended community set elements.
In the following example, a cost extended community set named extcomm-cost is defined:

```
RP/0/RP0/CPU0:router(config)# extcommunity-set cost extcomm-cost
RP/0/RP0/CPU0:router(config-ext)# IGP:90:914,
RP/0/RP0/CPU0:router(config-ext)# Pre-Bestpath:91:915
RP/0/RP0/CPU0:router(config-ext)# end-set
```
extcommunity-set rt

To define a Border Gateway Protocol (BGP) route target (RT) extended community set, use the extcommunity-set rt command in XR Config mode. To remove the RT community set, use the no form of this command.

```
extcommunity-set rt  name
no  extcommunity-set rt  name
```

**Syntax Description**
- `name` Name of an RT extended community set.

**Command Default**
No default behavior or values

**Command Modes**
XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `extcommunity-set rt` command to define an RT extended community set for BGP.

Regular expressions and ranges can be specified to match the extended communities. Regular expressions and ranges can be specified in an extended community set to support the matching of communities. An attempt to use an extended community set that contains a range or regular expression to set an extended community set value is rejected when an attempt to attach such a policy is made.

An extcommunity set RT holds RT extended community values to match against the Border Gateway Protocol (BGP) RT extended community attribute. RT extended communities can be entered in these formats:

- `#-remark` --- Remark beginning with '#'
- `*` --- Wildcard (any community or part thereof)
- `1-4294967295` --- 32-bit decimal number
- `1-65535` --- 16-bit decimal number
- `A.B.C.D/M:N` --- Extended community - IPv4 prefix format
- `A.B.C.D:N` --- Extended community - IPv4 format
- `ASN:N` --- Extended community - ASPLAIN format
- `X.Y:N` --- Extended community - ASDOT format
- `dfa-regex` --- DFA (deterministic finite automata) style regular expression
- `ios-regex` --- Traditional IOS style regular expression
The dfa-regex and ios-regex syntax for community set is \

```
["[^':&<> 
]*:[^':&<> 
]*"]
```

This means that regex starts with a single-quote ("), followed by a string of any character (that does not include single-quote, colon, ampersand, less-than, greater-than, or space) followed by a colon, and a string of any characters (that does not include single-quote, colon, ampersand, less-than, greater-than, or space) followed by single-quote.

\( N \) is a number within the range of 1 to 65535.

### Examples

In the following example, an RT extended community set named extcomm-rt is defined:

```
RP/0/RP0/CPU0:router(config)# extcommunity-set rt extcomm-rt
RP/0/RP0/CPU0:router(config-ext)# 10002:666
RP/0/RP0/CPU0:router(config-ext)# 10.0.0.2:666
RP/0/RP0/CPU0:router(config-ext)# end-set
```
extcommunity-set soo

To define a Border Gateway Protocol (BGP) Site-of-Origin (SoO) extended community set, use the `extcommunity-set soo` command in XR Config mode mode. To remove the SoO extended community set, use the `no` form of this command.

```
extcommunity-set soo name
no extcommunity-set soo name
```

**Syntax Description**

<table>
<thead>
<tr>
<th><strong>name</strong></th>
<th>Name of an SoO extended community set.</th>
</tr>
</thead>
</table>

**Command Default**

No default behavior or values

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `extcommunity-set soo` command to define an SoO extended community set.

An extcommunity set soo holds SoO extended community values to match against the Border Gateway Protocol (BGP) SoO extended community attribute. SoO extended communities can be entered in these formats:

- `#-remark` --- Remark beginning with '#'
- `*` --- Wildcard (any community or part thereof)
- `1-4294967295` --- 32-bit decimal number
- `1-65535` --- 16-bit decimal number
- `A.B.C.D/M:N` --- Extended community - IPv4 prefix format
- `A.B.C.D:N` --- Extended community - IPv4 format
- `ASN:N` --- Extended community - ASPLAIN format
- `X.Y:N` --- Extended community - ASDOT format
- `abort` --- Discard RPL definition and return to top level config
- `dfa-regex` --- DFA style regular expression
- `end-set` --- End of set definition
- `exit` --- Exit from the submode
- `ios-regex` --- Traditional IOS style regular expression
- `show` --- Show partial RPL configuration

*N* is a site-specific number.
In the following example, a SoO extended community set named extcomm-soo is defined:

```plaintext
RP/0/RP0/CPU0:router(config)# extcommunity-set soo extcomm-soo
RP/0/RP0/CPU0:router(config-ext)# 66:60001,
RP/0/RP0/CPU0:router(config-ext)# 77:70001,
RP/0/RP0/CPU0:router(config-ext)# 88:80001,
RP/0/RP0/CPU0:router(config-ext)# 99:90001,
RP/0/RP0/CPU0:router(config-ext)# 100.100.100.1:153
RP/0/RP0/CPU0:router(config-ext)# end-set
```
extcommunity soo is-empty

To determine if a Border Gateway Protocol (BGP) route has any Site-of-Origin (SoO) extended communities associated with it, use the `extcommunity soo is-empty` command in route-policy configuration mode.

**extcommunity soo is-empty**

**Syntax Description**
This command has no arguments or keywords.

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `extcommunity soo is-empty` command as a conditional expression within an `if` statement to check if a BGP SoO route has extended community attributes associated with it.

**Note**
For a list of all conditional expressions available within an `if` statement, see the `if` command.

The `is-empty` operator takes no arguments and evaluates to true if the route has no SoO extended community attributes associated with it.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
In the following example, if a route has no SoO extended communities associated with it, the local preference is set to 100:

```bash
RP/0/RP0/CPU0:router(config)# route-policy extcommunity-is-empty-example
RP/0/RP0/CPU0:router(config-rpl)# if extcommunity soo is-empty then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
extcommunity soo matches-any

To match any element of a Border Gateway Protocol (BGP) Site-of-Origin (SoO) extended community set, use the extcommunity soo matches-any command in route-policy configuration mode.

```
extcommunity soo matches-any {extcommunity-set-name inline-extcommunity-set parameter}
```

**Syntax Description**
- extcommunity-set-name: Name of a SoO extended community set.
- inline-extcommunity-set: Inline SoO extended community set. The inline extended community set must be enclosed in parentheses.
- parameter: Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**
- Release 6.0: This command was introduced.

**Usage Guidelines**
Use the `extcommunity soo matches-any` command as a conditional expression within an `if` statement to match elements of an extended community set.

**Note**
For a list of all conditional expressions available within an `if` statement, see the `if` command.

A simple condition using the `matches-any` operator evaluates as true if at least one extended community in the route matches an extended community specification in the named or inline set. If no extended community in the route matches any of the specifications in the named or inline set, then this simple condition evaluates to false. Likewise, when there is no extended community at all in the route, the condition evaluates to false.

Matching an extended community in the route to a specification in a named or an inline set is intuitive. In inline sets, extended community specifications may be parameterized, in which case the relevant matching is done when the value of the parameter has been supplied.

**Task ID**
- Task ID: Operations
  - route-policy: read, write

**Examples**
In the following example, an SoO extended community set named extcomm-soo and a parameterized route policy named my-extcommunity-set-example($tag,$ip) are defined.

The condition route policy named extcommunity soo matches-any is used in an `if` statement in this policy. If it evaluates to true, the local preference value is set to 100.
If it evaluates to false, the SoO extended community is evaluated using an inline set. If it evaluates to true, the local preference value is set to 200.

If it evaluates to false, the SoO extended community is evaluated using a different inline set. If it evaluates to true, the local preference value is set to 300.

If it evaluates to false, the SoO extended community is evaluated using a different inline set. If it evaluates to true, the local preference value is set to 400.

```
RP/0/RP0/CPU0:router(config)# extcommunity-set soo extcomm-soo
RP/0/RP0/CPU0:router(config-ext)# 66:60001,
RP/0/RP0/CPU0:router(config-ext)# 77:70001,
RP/0/RP0/CPU0:router(config-ext)# 88:80001,
RP/0/RP0/CPU0:router(config-ext)# 99:90001,
RP/0/RP0/CPU0:router(config-ext)# 100.100.100.1:153
RP/0/RP0/CPU0:router(config-ext)# end-set

RP/0/RP0/CPU0:router(config)# route-policy my-extcommunity-set-example($tag,$ip)
RP/0/RP0/CPU0:router(config-rpl)# if extcommunity soo matches-any extcomm-soo then
RP/0/RP0/CPU0:router(config-rpl-if)# elseif extcommunity soo matches-any (10:20, 10:$tag) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif extcommunity soo matches-any ($ip:$tag) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif extcommunity soo matches-any (2.3.4.5:$tag) then
RP/0/RP0/CPU0:router(config-rpl-elseif)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
extcommunity soo matches-every

To match every element of a Border Gateway Protocol (BGP) Site-of-Origin (SoO) extended community set, use the `extcommunity soo matches-every` command in route-policy configuration mode.

**extcommunity soo matches-every** `{extcommunity-set-nameinline-extcommunity-setparameter}`

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>extcommunity-set-name</strong></td>
<td>Name of a SoO extended community set.</td>
</tr>
<tr>
<td><strong>inline-extcommunity-set</strong></td>
<td>Inline SoO extended community set. The inline extended community set must be enclosed in parentheses.</td>
</tr>
<tr>
<td><strong>parameter</strong></td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
</tbody>
</table>

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `extcommunity soo matches-every` command as a conditional expression within an `if` statement to match every element of a SoO extended community set.

**Note**

For a list of all conditional expressions available within an `if` statement, see the `if` command.

A simple condition using the `matches-every` operator evaluates as true if every extended community value in the extended community attribute for the route matches at least one element of the extended community set or inline set. If no extended community in the route matches any of the specifications in the named or inline set, then this simple condition evaluates to false. Likewise, when there is no extended community at all in the route, the condition evaluates to false.

Matching an extended community in the route to a specification in a named or an inline set is intuitive. In inline sets, extended community specifications may be parameterized, in which case the relevant matching is done when the value of the parameter has been supplied.

**Examples**

In the following example, an extended community set named `my-extcomm-rt-set` and a parameterized route policy named `extcommunity-matches-every-example($as, $tag)` are defined. The condition `extcommunity soo matches-every` is used in an `if` statement in this policy and if it evaluates to true, the local-preference value is set to 100. If it evaluates to false, the extended community is evaluated using an inline set. If that condition evaluates to true, the local-preference value is set to 200. If it evaluates to false, the local-preference value is set to 300.

```
RP/0/RP0/CPU0:router (config)# extcommunity-set soo my-extcomm-rt-set
RP/0/RP0/CPU0:router(config-ext)# 10:20,
RP/0/RP0/CPU0:router(config-ext)# 10:30,
```
RP/0/RP0/CPU0:router (config-ext) # 10:40
RP/0/RP0/CPU0:router (config-ext) # end-set

RP/0/RP0/CPU0:router (config) # route-policy extcommunity-matches-every-example ($as, $tag)
RP/0/RP0/CPU0:router (config-rpl) # if extcommunity soo matches-every my-extcomm-rt-set then
RP/0/RP0/CPU0:router (config-rpl-if) # set local-preference 100
RP/0/RP0/CPU0:router (config-rpl-if) # elseif extcommunity soo matches-every (10:20, 10:$tag, $as:30) then
RP/0/RP0/CPU0:router (config-rpl-elseif) # set local-preference 200
RP/0/RP0/CPU0:router (config-rpl-elseif) # else
RP/0/RP0/CPU0:router (config-rpl-elseif) # set local-preference 300
RP/0/RP0/CPU0:router (config-rpl-elseif) # endif
RP/0/RP0/CPU0:router (config-rpl) # end-policy
To decide which actions or dispositions should be taken for a given route, use the `if` command in route-policy configuration mode.

```
if conditional-expression then action-statement [action-statement] [elseif conditional-expression then action-statement [action-statement]] [else action-statement [action-statement]] endif
```

### Syntax Description
- **conditional-expression**: Expression to decide which actions or dispositions should be taken for the given route.
- **then**: Executes an action statement if the `if` condition is true.
- **elseif**: Strings together a sequence of tests.
- **else**: Executes an action statement if the `if` condition is false.
- **endif**: Ends the `if` statement.
- **action-statement**: Sequence of operations that modify a route.

### Command Default
No default behavior or values

### Command Modes
Route-policy configuration

### Command History
**Release** 6.0  This command was introduced.

### Usage Guidelines
The `if` command uses a conditional expression to decide which actions or dispositions should be taken for a given route. Table 108: Conditional Expressions, on page 609 lists the conditional expressions.

An action statement is a sequence of operations that modify a route, most of which are distinguished by the `set` keyword. In a route policy, these operations can be grouped. Table 109: Action Statements, on page 610 lists the action statements.

Apply Condition policies allow usage of a route-policy in an "if" statement of another route-policy.

```
Route-policy policy_name
If apply policyA and apply policyB then
  Set med 100
Else if not apply policyD then
  Set med 200
Else
  Set med 300
Endif
End-policy
```
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>as-path in</code>, on page 552</td>
<td>Matches the AS path of a route to an AS path set. The AS path is a sequence of autonomous system numbers traversed by a route.</td>
</tr>
<tr>
<td><code>as-path is-local</code>, on page 554</td>
<td>Determines if the router (or another router within this autonomous system or confederation) originated the route.</td>
</tr>
<tr>
<td><code>as-path length</code>, on page 555</td>
<td>Performs a conditional check based on the length of the AS path.</td>
</tr>
<tr>
<td><code>as-path neighbor-is</code>, on page 556</td>
<td>Tests the autonomous system number or numbers at the head of the AS path against a sequence of one or more integral values or parameters.</td>
</tr>
<tr>
<td><code>as-path originates-from</code>, on page 558</td>
<td>Tests an AS path against the AS sequence beginning with the AS number that originated a route.</td>
</tr>
<tr>
<td><code>as-path passes-through</code>, on page 560</td>
<td>Tests to learn if the specified integer or parameter appears anywhere in the AS path or if the sequence of integers and parameters appears.</td>
</tr>
<tr>
<td><code>as-path unique-length</code>, on page 564</td>
<td>Performs specific checks based on the length of the AS path.</td>
</tr>
<tr>
<td><code>community is-empty</code>, on page 566</td>
<td>Learns if a route has community attributes associated with it.</td>
</tr>
<tr>
<td><code>community matches-any</code>, on page 567</td>
<td>Matches any element of a community set.</td>
</tr>
<tr>
<td><code>community matches-every</code>, on page 569</td>
<td>Matches every element of a community set.</td>
</tr>
<tr>
<td><code>destination in</code>, on page 578</td>
<td>Matches a destination entry in a named prefix set or inline prefix set.</td>
</tr>
<tr>
<td><code>extcommunity rt is-empty</code>, on page 590</td>
<td>Learns if a route has RT extended community attributes associated with it.</td>
</tr>
<tr>
<td><code>extcommunity rt matches-any</code>, on page 591</td>
<td>Matches elements of an RT extended community set.</td>
</tr>
<tr>
<td><code>extcommunity rt matches-every</code>, on page 593</td>
<td>Matches every element of an RT extended community set.</td>
</tr>
<tr>
<td><code>extcommunity rt matches-within</code>, on page 595</td>
<td>Matches at least one element of a Border Gateway Protocol (BGP) route target (RT) extended community set.</td>
</tr>
<tr>
<td><code>extcommunity soo is-empty</code>, on page 603</td>
<td>Learns if a route has SoO extended community attributes associated with it.</td>
</tr>
<tr>
<td><code>extcommunity soo matches-any</code>, on page 604</td>
<td>Matches elements of an SoO extended community set.</td>
</tr>
<tr>
<td><code>extcommunity soo matches-every</code>, on page 606</td>
<td>Matches every element of an SoO extended community set.</td>
</tr>
</tbody>
</table>
### Command Language Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>local-preference, on page 618</td>
<td>Specifies BGP local-preference attribute</td>
</tr>
<tr>
<td>med, on page 619</td>
<td>Compares the MED to an integer value or a parameterized value.</td>
</tr>
<tr>
<td>next-hop in, on page 620</td>
<td>Compares the next-hop associated with the route to data contained in either a named or an inline prefix set.</td>
</tr>
<tr>
<td>orf prefix in, on page 621</td>
<td>Matches a prefix in a prefix set or an inline prefix set.</td>
</tr>
<tr>
<td>origin is, on page 623</td>
<td>Tests the value of the origin attribute.</td>
</tr>
<tr>
<td>path-type is, on page 628</td>
<td>Tests the path type.</td>
</tr>
<tr>
<td>protocol, on page 634</td>
<td>Checks if a protocol is installing the route.</td>
</tr>
<tr>
<td>rd in, on page 636</td>
<td>Compares the RD associated with the route to data contained in either a named or an inline RD set.</td>
</tr>
<tr>
<td>rib-has-route, on page 642</td>
<td>Checks if a route is in the RIB.</td>
</tr>
<tr>
<td>route-has-label, on page 643</td>
<td>Checks if a route has a Multiprotocol Label Switching (MPLS) label.</td>
</tr>
<tr>
<td>route-type is, on page 646</td>
<td>Compares route types when redistribution is being performed into BGP, OSPF, or IS-IS.</td>
</tr>
<tr>
<td>source in, on page 773</td>
<td>Tests the source of the route against the data in either a named or an inline prefix set.</td>
</tr>
<tr>
<td>tag, on page 776</td>
<td>Matches a specific tag value.</td>
</tr>
<tr>
<td>tag in, on page 777</td>
<td>Conditionally compares tag-route against tag-set.</td>
</tr>
<tr>
<td>vpn-distinguisher is, on page 781</td>
<td>Compares the VPN distinguisher against a specified value.</td>
</tr>
</tbody>
</table>

### Action Statements

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>abort (RPL), on page 547</td>
<td>Discards a route policy definition and returns to XR Config mode.</td>
</tr>
<tr>
<td>add, on page 549</td>
<td>Adds an offset to an existing value.</td>
</tr>
<tr>
<td>apply, on page 550</td>
<td>Executes a parameterized or an unparameterized policy from within another policy.</td>
</tr>
<tr>
<td>delete community, on page 574</td>
<td>Deletes community values from a community list in a route.</td>
</tr>
<tr>
<td>delete extcommunity rt, on page 576</td>
<td>Deletes extended community values from an extended community list in a route.</td>
</tr>
<tr>
<td>done, on page 580</td>
<td>Accepts this route with no further processing</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>drop, on page 582</td>
<td></td>
</tr>
<tr>
<td>end-policy, on page 587</td>
<td></td>
</tr>
<tr>
<td>pass, on page 627</td>
<td></td>
</tr>
<tr>
<td>prepend as-path, on page 632</td>
<td></td>
</tr>
<tr>
<td>replace as-path, on page 639</td>
<td></td>
</tr>
<tr>
<td>set community, on page 653</td>
<td></td>
</tr>
<tr>
<td>set dampening, on page 656</td>
<td></td>
</tr>
<tr>
<td>set extcommunity cost, on page 658</td>
<td></td>
</tr>
<tr>
<td>set extcommunity rt, on page 660</td>
<td></td>
</tr>
<tr>
<td>set ip-precedence, on page 662</td>
<td></td>
</tr>
<tr>
<td>set isis-metric, on page 663</td>
<td></td>
</tr>
<tr>
<td>set label, on page 664</td>
<td></td>
</tr>
<tr>
<td>set level, on page 667</td>
<td></td>
</tr>
<tr>
<td>set local-preference, on page 668</td>
<td></td>
</tr>
<tr>
<td>set med, on page 669</td>
<td></td>
</tr>
<tr>
<td>set metric-type (IS-IS), on page 671</td>
<td></td>
</tr>
<tr>
<td>set metric-type (OSPF), on page 673</td>
<td></td>
</tr>
<tr>
<td>set next-hop, on page 674</td>
<td></td>
</tr>
<tr>
<td>set origin, on page 676</td>
<td></td>
</tr>
<tr>
<td>set ospf-metric, on page 677</td>
<td></td>
</tr>
<tr>
<td>set qos-group (RPL), on page 679</td>
<td></td>
</tr>
<tr>
<td>set rib-metric, on page 680</td>
<td></td>
</tr>
<tr>
<td>set rip-metric, on page 681</td>
<td></td>
</tr>
<tr>
<td>set rip-tag, on page 682</td>
<td></td>
</tr>
<tr>
<td>set tag, on page 686</td>
<td></td>
</tr>
</tbody>
</table>
**Command** | **Description**
--- | ---
set traffic-index, on page 687 | Sets the traffic index attribute.
set weight, on page 690 | Sets the weight value for BGP routes.
suppress-route, on page 775 | Indicates that a given component of an aggregate should be suppressed, that is, not advertised.
unsuppress-route, on page 780 | Indicates that a given component of an aggregate should be unsuppressed.
set vpn-distinguisher, on page 689 | Sets the VPN distinguisher value.

**Task ID**
**Task ID** | **Operations**
--- | ---
route-policy | read, write

### Examples

In the following example, any route whose AS path is in the set as-path-set-1 is dropped:

```
RP/0/RP0/CPU0:router(config-rpl)# if as-path in as-path-set-1 then
RP/0/RP0/CPU0:router(config-rpl-if)# drop
RP/0/RP0/CPU0:router(config-rpl-if)# endif
```

The contents of the **then** clause may be an arbitrary sequence of action statements.

The following example shows an **if** statement with two action statements:

```
RP/0/RP0/CPU0:router(config-rpl)# if origin is igp then
RP/0/RP0/CPU0:router(config-rpl-if)# set med 42
RP/0/RP0/CPU0:router(config-rpl-if)# prepend as-path 73 5
RP/0/RP0/CPU0:router(config-rpl-if)# endif
```

The **if** command also permits an **else** clause to be executed if the expression is false, as follows:

```
RP/0/RP0/CPU0:router(config-rpl)# if med eq 200 then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 10
RP/0/RP0/CPU0:router(config-rpl-if)# else
RP/0/RP0/CPU0:router(config-rpl-else)# set community (12:34) additive
RP/0/RP0/CPU0:router(config-rpl-else)# endif
```

The routing policy language (RPL) also provides syntax using the **elseif** command to string together a sequence of tests, as shown in the following example:

```
RP/0/RP0/CPU0:router(config-rpl)# if med eq 150 then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 10
RP/0/RP0/CPU0:router(config-rpl-if)# elseif med eq 200 then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 60
```
The statements within an if statement may themselves be if statements, as shown in this example:

```plaintext
RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif med eq 250 then
RP/0/RP0/CPU0:router(config-rpl-elseif)# set local-preference 110
RP/0/RP0/CPU0:router(config-rpl-elseif)# else
RP/0/RP0/CPU0:router(config-rpl-else)# set local-preference 0
RP/0/RP0/CPU0:router(config-rpl-else)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```

The policy configuration shown sets the value of the local preference attribute to 100 on any route that has a community value of 12:34 or 56:78 associated with it. However, if any of these routes has a Multi Exit Discriminator (MED) value of 150, then each route with both the community value of 12:34 or 56:78 and a MED of 150 is dropped.

```plaintext
RP/0/RP0/CPU0:router(config-rpl)# if community matches-any (12:34, 56:78) then
RP/0/RP0/CPU0:router(config-rpl-if)# if med eq 150 then
RP/0/RP0/CPU0:router(config-rpl-if)# drop
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
if route-aggregated

To match the aggregated routes from the other routes, use the if route-aggregated command in route policy configuration mode.

Syntax Description

<table>
<thead>
<tr>
<th>Syntax Description</th>
</tr>
</thead>
</table>
| route-aggregated   | Checks if route is an aggregation of multiple routes.

Command Default

No default behavior or values

Command Modes

Route-policy configuration

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| Release 6.0 | This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples

This example shows how to match the aggregated routes from other routes:

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# route-policy route-policy atomic_agg
RP/0/RP0/CPU0:router(config-rpl)# if route-aggregated then
RP/0/RP0/CPU0:router(config-rpl-if)# set extcommunity rt (1:1)
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
is-best-path

To tag the path selected as the best path use the **is-best-path** command in route policy configuration mode.

**is-best-path**

**Syntax Description**

- **is-best-path** Checks and tags the path selected as best-path.

**Command Default**

No default behavior or values.

**Command Modes**

Route-policy configuration

**Command History**

- **Release** Modification
  - Release 6.0 This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Example**

```
RP/0/RSP0RP00/CPU0:router(config)# route-policy
RP/0/RSP0RP00/CPU0:router(config)# route-policy sample
RP/0/RSP0RP00/CPU0:router(config-rpl)# if destination i in is-backup-path is-best-external is-best-path
  if destination is-best-path then
    set community community
  endif
end-policy
```

```
RP/0/RSP0RP00/CPU0:router# sh version
Wed Jul 8 16:08:34.286 IST
Cisco IOS XR Software, Version 5.3.2.147[EnXR]
Copyright (c) 2015 by Cisco Systems, Inc.
Built on Fri Jun 26 17:35:16 IST 2015
By router in RP/0/RSP0RP00/CVO
```
is-backup-path

To tag all the paths equal to the backup path use, **is-backup-path** command in route policy configuration mode.

<table>
<thead>
<tr>
<th><strong>Syntax Description</strong></th>
<th><strong>is-backup-path</strong> Checks and tags the path selected as backup path.</th>
</tr>
</thead>
</table>

**Command Default**
No default behavior or values.

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th><strong>Release</strong></th>
<th><strong>Modification</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th><strong>Task ID</strong></th>
<th><strong>Operation</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Example**

```
RP/0/RSP0RP00/CPU0:router(config)# route-policy WORD Route Policy name
RP/0/RSP0RP00/CPU0:router(config)# route-policy sample
RP/0/RSP0RP00/CPU0:router(config-rpl)# if destination in is-backup-path is-best-external is-best-path

RP/0/RSP0RP00/CPU0:router(config)# route-policy WORD Route Policy name
RP/0/RSP0RP00/CPU0:router(config)# route-policy sample
RP/0/RSP0RP00/CPU0:router(config-rpl)# if destination in is-backup-path is-best-external is-best-path
```
**is-multi-path**

To tag all the paths equal to the best path based on multi-path context use, **is-multi-path** command in route policy configuration mode.

**Syntax Description**

```
is-multi-path  Checks and tag all the path equal to the as best-path.
```

**Command Default**

No default behavior or values.

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Example**

```
RP/0/RSP0RP00/CPU0:router(config)#route-policy
WORD  Route Policy name
RP/0/RSP0RP00/CPU0:router(config)#route-policy sample
RP/0/RSP0RP00/CPU0:router(config-rpl)#if destination in
   is-backup-path is-best-external is-best-path
is-multi-path
RP/0/RSP0RP00/CPU0:router(config-rpl)#if destination is-
   is-backup-path is-best-external is-best-path is-multi-path
RP/0/RSP0RP00/CPU0:router(config-rpl)#if destination is-best-path then
RP/0/RSP0RP00/CPU0:router(config-rpl-if)#set l
   label label-index label-mode level
   community lsm-root
RP/0/RSP0RP00/CPU0:router(config-rpl-if)#set community community
RP/0/RSP0RP00/CPU0:router(config-rpl-if)#endif
RP/0/RSP0RP00/CPU0:router(config-rpl)#end-policy
RP/0/RSP0RP00/CPU0:router(config)#commit
Wed Jul  8 16:08:23.436 IST
```
**local-preference**

To compare the local-preference attribute of a BGP route to an integer value or a parameterized value, use the local-preference command in route-policy configuration mode.

```
local-preference {eq|is|ge|le} {number|parameter}
```

**Syntax Description**

- **eq | is | ge | le** Equal to; exact match; greater than or equal to; less than or equal to.
- **number** Value assigned to a 32-bit unsigned integer. Range is 0 to 4294967295.
- **parameter** Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **local-preference** command as a conditional expression within an **if** statement to compare the local-preference attribute to an integer value or a parameterized value.

**Note**

For a list of all conditional expressions available within an **if** statement, see the **if** command.

The MED is a 32-bit unsigned integer. The **eq** operation compares the local-preference to either a static value or a parameterized value passed to a parameterized policy for equality with that value. A greater than or equal to comparison can also be done with the **ge** operator, and a less than or equal to comparison can be performed using the **le** operator.

**Examples**

The following example shows that if the **local-preference** is 10, local-preference is set to 100:

```
RP/0/RSP0R0/CP00:router(config-rpl)# if local-preference eq 10 then
RP/0/RSP0R0/CP00:router(config-rpl-if)# set weight 100
RP/0/RSP0R0/CP00:router(config-rpl-if)# endif
```

```
RP/0/RSP0R0/CP00:router(config-rpl)#
```
med

To compare the Multi Exit Discriminator (MED) to an integer value or a parameterized value or compare the MED attribute of a BGP route to an integer value, use the `med` command in route-policy configuration mode.

```plaintext
med {eq|is|ge|le} {number|parameter}
```

**Syntax Description**

- `eq | is | ge | le` Equal to; exact match; greater than or equal to; less than or equal to.
- `number` Value assigned to a 32-bit unsigned integer. Range is 0 to 4294967295.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `med` command as a conditional expression within an `if` statement to compare the MED to an integer value or a parameterized value.

For a list of all conditional expressions available within an `if` statement, see the `if` command.

The MED is a 32-bit unsigned integer. The `eq` operation compares the MED to either a static value or a parameterized value passed to a parameterized policy for equality with that value. A greater than or equal to comparison can also be done with the `ge` operator, and a less than or equal to comparison can be performed using the `le` operator.

**Examples**

The following example shows that if the `med` commands match, the local preference is set to 100:

```plaintext
RP/0/RP0/CPU0:router(config-rpl)# if med eq 10 then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
next-hop in

To compare the next-hop associated with the route to data contained in either an inline or a named prefix set, use the `next-hop in` command in route-policy configuration mode.

```
next-hop in  {prefix-set-name|inline-prefix-set|parameter}
```

**Syntax Description**

- `prefix-set-name` Name of a prefix set.
- `inline-prefix-set` Inline prefix set. The inline prefix set must be enclosed in parentheses.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `next-hop in` command as a conditional expression within an `if` statement to compare the next-hop associated with the route to data contained in either an inline or a named prefix set. The result is true if any value in the prefix set matches the next-hop of the route. A comparison that refers to a named prefix set that has no elements in it returns false.

**Note**

For a list of all conditional expressions available within an `if` statement, see the `if` command.

The next-hop is an IPv4 address entered as a dotted-decimal or an IPv6 address entered as a colon-separated hexadecimal.

**Examples**

The following example shows that if the `next-hop in` commands match, the local preference is set to 100

```
RP/0/RP0/CPU0:router(config-rpl)# if next-hop in some-prefix-set then
RP/0/RP0/CPU0:router(config-rpl-if)# if next-hop in (10.0.0.5, fe80::230/64) then
RP/0/RP0/CPU0:router(config-rpl-if)# set local-preference 0
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
orf prefix in

To configure an outbound route filter (ORF), use the `orf prefix in` command in route-policy configuration mode.

```
orf prefix in {prefix-set-name|inline-prefix-set}
```

**Syntax Description**
- `prefix-set-name` Name of a prefix set.
- `inline-prefix-set` Inline prefix set. The inline prefix set must be enclosed in parentheses.

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Use the `orf prefix in` command to match a prefix in a prefix set or an inline prefix set.

This command takes either a named prefix set or an inline prefix set value as an argument. It returns true if the destination NLRI matches any entry in the prefix set. An attempt to match destination using a prefix set that is defined but contains no elements returns false.

This command is used in the context of the `orf` route-policy attach point in BGP. The destination of a route is also known in Border Gateway Protocol (BGP) as its network-layer reachability information (NLRI). It comprises a prefix value and a mask length. The routing policy language (RPL) provides one operation on prefixes, testing them for matching against a list of prefix-match specifications using the `in` operator.

**Examples**
In the following example, the prefix set `orfpreset1` and the route policy named `orfpolicy` are defined. Next, the `orfpolicy` is applied to the neighbor `orf` attach point.

If the prefix of the route matches any of the prefixes specified in `orfpreset1` (211.105.1.0/24, 211.105.5.0/24, 211.105.11.0/24), then the prefix is dropped. If the prefix matches `in(211.105.3.0/24, 211.105.7.0/24, 211.105.13.0/24)`, then the prefix is accepted. In addition to this inbound filtering, BGP sends these prefix entries to the upstream neighbor indicating a permit or deny so that the neighbor can make the same filter updates.

```
RP/0/RP0/CPU0:router(config)# prefix-set orfpreset1
RP/0/RP0/CPU0:router(config-pfx)# 211.105.1.0/24,
RP/0/RP0/CPU0:router(config-pfx)# 211.105.5.0/24,
RP/0/RP0/CPU0:router(config-pfx)# 211.105.11.0/24
RP/0/RP0/CPU0:router(config-pfx)# end-set
!
!
RP/0/RP0/CPU0:router(config)# route-policy orfpolicy
RP/0/RP0/CPU0:router(config-rpl)# if orf prefix in orfpreset1 then
RP/0/RP0/CPU0:router(config-rpl-if)# drop
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# if orf prefix in (211.105.3.0/24, 211.105.7.0/24,
```
211.105.13.0/24) then
RP/0/RP0/CPU0:router(config-rpl-if)# pass
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
!
RP/0/RP0/CPU0:router(config)# router bgp 2
RP/0/RP0/CPU0:router(config-bgp)# neighbor 1.1.1.1
RP/0/RP0/CPU0:router(config-bgp-nbr)# remote-as 3
RP/0/RP0/CPU0:router(config-bgp-nbr)# address-family ipv4 unicast
RP/0/RP0/CPU0:router(config-bgp-nbr-af)# orf route-policy orfpolicy
**origin is**

To match a specific origin type, use the **origin is** command in route-policy configuration mode.

**origin is  
\{igp\}|egp|incomplete\}**

**Syntax Description**
- **igp**  
  Specifies Interior Gateway Protocol.
- **egp**  
  Specifies Exterior Gateway Protocol.
- **incomplete**  
  Specifies that Border Gateway Protocol (BGP) first learned the route by means other than BGP or Interior Gateway Protocol (IGP); for example, the route is learned through configuration.

---

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the **origin is** command as a conditional expression within an **if** statement to test the value of the origin attribute.

---

**Note**
For a list of all conditional expressions available within an **if** statement, see the **if** command.

The origin of a BGP route is an enumeration; it is **igp**, **egp**, or **incomplete**. This command can be parameterized.

**Examples**

In the following example, the origin is tested within an **if** statement to learn if it is either **igp** or **egp**:

```
RP/0/RP0/CPU0:router(config-rpl)# if origin is igp or origin is egp then
```

In the following example, a parameter is used to match a specific origin type:

```
RP/0/RP0/CPU0:router(config)# route-policy bar($origin)
RP/0/RP0/CPU0:router(config-rpl)# if origin is $origin then
```
RP/0/RP0/CPU0:router(config-rpl-if)# set med 20
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
ospf-area

To match a specific ospf area, use the `ospf-area` command in route-policy configuration mode.

ospf-area [all-paths] {in|is}

**Syntax Description**

- **is**: Specify the explicit `area-id`.
- **in**: Specify a list of `area-id` or `area-set`. Multiple areas can be specified separated by a comma (,).
- **all-paths**: Used for routes with multiple paths. A match is made if area for every path of the route is configured in the route-policy.

**Command Default**

None

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The route policy define by using `ospf-area` is useful in redistributing routes from a specific area of a routing domain into OSPF. After the route policy is created, use the `redistribute ospf route-policy` command for route redistribution.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Example**

In the following example, an explicit area is specified as the matching criteria.

```
RP/0/RP0/CPU0:router(config-rpl)# if ospf-area is 10 then pass else drop endif
```

In the following example, a collection of areas is specified as the matching criteria.

```
RP/0/RP0/CPU0:router(config-rpl)# if ospf-area in (5,6,255.255.10.2) then pass else drop endif
```

In the following example, an area set is specified as the matching criteria. As a pre-requisite, the area set must be defined.

```
RP/0/RP0/CPU0:router(config)# ospf-area-set S1
RP/0/RP0/CPU0:router(config-ospf-area)# 1, 2.2.2.2 end-set
RP/0/RP0/CPU0:router(config)# route-policy P1
```
RP/0/RP0/CPU0:router(config-rpl)# if ospf-area in S1 then pass else drop endif
pass

To pass a route for further processing, use the pass command in route-policy configuration mode.

pass

Syntax Description

This command has no arguments or keywords.

Command Default

No default behavior or values

Command Modes

Route-policy configuration

Command History

Release Modification

Release 6.0 This command was introduced.

Usage Guidelines

Use the pass command to signify that even though this route has not been modified, the user wants to continue executing in this policy block.

The pass command can be used as an action statement within an if statement. For a list of all action statements available within an if statement, see the if command.

When a policy block has finished executing, any route that has been modified in this policy block or has received a pass disposition in this policy block passes the policy and execution finishes for that policy. If this policy block is applied from within another policy block and the route is either passed or modified, then execution continues in the policy block that applied this policy block.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples

The following example shows how to accept the route unconditionally without modifying it:

RP/0/RP0/CPU0:router(config-rpl)# pass

This example accepts the route unconditionally, without modifying it, if the destination is in prefix-set permitted:

RP/0/RP0/CPU0:router(config-rpl)# if destination in permitted then
RP/0/RP0/CPU0:router(config-rpl-if)# pass
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
path-type is

To match path types, use the **path-type is** command in route-policy configuration mode.

```
path-type is {ibgp|ebgp}parameter
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ibgp</td>
<td>Specifies an internal BGP path.</td>
</tr>
<tr>
<td>egp</td>
<td>Specifies an external BGP path.</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **path-type is** command as a conditional expression within an **if** statement to match path types.

For a list of all conditional expressions available within an **if** statement, see the **if** command.

**Examples**

In the following example, if the path is an external BGP path the route is accepted:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_A
RP/0/RP0/CPU0:router(config-rpl)# if path-type is ebgp then
RP/0/RP0/CPU0:router(config-rpl-if)# pass
RP/0/RP0/CPU0:router(config-rpl-if)# else
RP/0/RP0/CPU0:router(config-rpl-else)# drop
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**policy-global**

To define global parameters and enter global parameter configuration mode, use the **policy-global** command in XR Config mode. To remove global parameters, use the no form of this command.

```
    policy-global
    no policy-global
```

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

No default behavior or values

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **policy-global** command to define global parameters and enter global parameter configuration mode.

RPL supports the definition of systemwide global parameters that can be used inside a policy definition. The global parameter values can be used directly inside a policy definition similar to the local parameters of parameterized policy. When a parameterized policy has a parameter name “collision” with a global parameter name, parameters local to policy definition take precedence, effectively ‘masking off’ global parameters. In addition, a validation mechanism is in place to prevent the deletion of a particular global parameter if it is referred by any policy. For more information on global parameters and parameterization, see the Implementing Routing Policy module of the Routing Configuration Guide for Cisco NCS 5000 Series Routers.

**Examples**

The following example shows how to configure global parameters:

```
RP/0/RP0/CPU0:router(config)# policy-global
RP/0/RP0/CPU0:router(config-rp-gl)# glbpathtype 'ebgp'
RP/0/RP0/CPU0:router(config-rp-gl)# glbtag '100'
RP/0/RP0/CPU0:router(config-rp-gl)# end-global
```

In the following example, the **globalparam** argument makes use of the global parameters gbpathtype and glbtag defined above and is defined for a nonparameterized policy:

```
RP/0/RP0/CPU0:router(config)# route-policy globalparam
RP/0/RP0/CPU0:router(config-rppl)# if path-type is $gbpathtype then
RP/0/RP0/CPU0:router(config-rppl)# set tag $glbtag
RP/0/RP0/CPU0:router(config-rppl)# endif
RP/0/RP0/CPU0:router(config-rppl)# end-policy
```
prefix-set

To enter prefix set configuration mode and define a prefix set for contiguous and non-contiguous set of bits, use the `prefix-set` command in XR Config mode. To remove a named prefix set, use the `no` form of this command.

```
prefix-set name
no prefix-set name
```

**Syntax Description**

- **name**
  - Name of a prefix set.

**Command Default**

None

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `prefix-set` command to enter prefix set configuration mode and define a prefix set.

A prefix set is a comma-separated list of prefix match specifications. It holds IPv4 or IPv6 prefix match specifications, each of which has four parts: an address, a mask length, a minimum matching length, and a maximum matching length. The address is required, but the other three parts are optional. The address is a standard four-part, dotted-decimal numeric IPv4 address or a colon-separated hexadecimal IPv6 address. The mask length, if present, is a nonnegative decimal integer in the range from 0 to 32 for IPv4 prefixes or 0 to 128 for IPv6 prefixes following the address and separated from it by a slash. The optional minimum matching length follows the address and optional mask length and is expressed as the keyword `ge` (mnemonic for greater than or equal to), followed by a nonnegative decimal integer in the range from 0 to 32 for IPv4 or 0 to 128 for IPv6. The optional maximum matching length follows the rest and is expressed by the keyword `le` (mnemonic for less than or equal to), followed by yet another nonnegative decimal integer in the range from 0 to 32 for IPv4 or 0 to 128 for IPv6. A syntactic shortcut for specifying an exact length for prefixes to match is the `eq` keyword, mnemonic for equal to.

If a prefix match specification has no mask length, then the default mask length is 32 for IPv4 or 128 for IPv6. The default minimum matching length is the mask length. If a minimum matching length is specified, then the default maximum matching length must be less than 32 for IPv4 prefixes or 128 for IPv6 prefixes. Otherwise, if neither a minimum nor maximum length is specified, the default maximum length is the mask length.

A prefix set is a list of prefix match specifications. It holds IPv4 or IPv6 prefix match specifications, each of which has two parts: an address and a mask. The address and mask is a standard dotted-decimal IPv4 or colon-separated hexadecimal IPv6 address. The prefix set allows the specifying of contiguous and non-contiguous set of bits that must be matched in any route. The set of bits to be matched are provided in the form of a mask in which a binary 0 means a mandatory match and a binary 1 means a 'do not match' condition.

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>
Examples

The following example shows a prefix set named legal-ipv4-prefix-examples:

```
RP/0/RP0/CPU0:router (config)# prefix-set legal-ipv4-prefix-examples
RP/0/RP0/CPU0:router (config-pfx)# 10.0.1.1,
RP/0/RP0/CPU0:router (config-pfx)# 10.0.2.0/24,
RP/0/RP0/CPU0:router (config-pfx)# 10.0.3.0/24 ge 28,
RP/0/RP0/CPU0:router (config-pfx)# 10.0.4.0/24 le 28,
RP/0/RP0/CPU0:router (config-pfx)# 10.0.5.0/24 ge 26 le 30,
RP/0/RP0/CPU0:router (config-pfx)# 10.0.6.0/24 eq 28
RP/0/RP0/CPU0:router (config-pfx)# end-set
```

The first element of the prefix set matches only one possible value, 10.0.1.1/32 or the host address 10.0.1.1. The second element matches only one possible value, 10.0.2.0/24. The third element matches a range of prefix values, from 10.0.3.0/28 to 10.0.3.255/32. The fourth element matches a range of values, from 10.0.4.0/24 to 10.0.4.240/28. The fifth element matches prefixes in the range from 10.0.5.0/26 to 10.0.5.252/30. The sixth element matches any prefix of length 28 in the range from 10.0.6.0/28 through 10.0.6.240/28.

The following prefix set consists entirely of invalid prefix match specifications:

```
RP/0/RP0/CPU0:router (config)# prefix-set INVALID-PREFIX-EXAMPLES
RP/0/RP0/CPU0:router (config-pfx)# 10.1.1.1 ge 16,
RP/0/RP0/CPU0:router (config-pfx)# 10.1.2.1 le 16,
RP/0/RP0/CPU0:router (config-pfx)# 10.1.3.0/24 le 23,
RP/0/RP0/CPU0:router (config-pfx)# 10.1.4.0/24 ge 33,
RP/0/RP0/CPU0:router (config-pfx)# 10.1.5.0/25 ge 29 le 28
RP/0/RP0/CPU0:router (config-pfx)# end-set
```

Neither the minimum length nor the maximum length is legal without a mask length. The maximum length must be at least the mask length. The minimum length must be less than 32, the maximum length of an IPv4 prefix. The maximum length must be equal to or greater than the minimum length.

The following example shows a valid IPv6 prefix set named legal-ipv6-prefix-examples:

```
RP/0/RP0/CPU0:router (config)# prefix-set legal-ipv6-prefix-examples
RP/0/RP0/CPU0:router (config-pfx)# 2001:0:0:1::/64,
RP/0/RP0/CPU0:router (config-pfx)# 2001:0:0:2::/64,
RP/0/RP0/CPU0:router (config-pfx)# 2001:0:0:3::/64,
RP/0/RP0/CPU0:router (config-pfx)# 2001:0:0:4::/64
RP/0/RP0/CPU0:router (config-pfx)# end-set
```

This example shows a prefix set named legal-ipv4-prefix:

```
RP/0/RP0/CPU0:router (config)# prefix-set legal-ipv4-prefix
RP/0/RP0/CPU0:router (config-pfx)# 10.1.1.1 0.255.0.255
RP/0/RP0/CPU0:router (config-pfx)# 10.2.2.2 0.0.0.0
RP/0/RP0/CPU0:router (config-pfx)# 10.3.3.3 255.255.255.254
RP/0/RP0/CPU0:router (config-pfx)# 10.4.4.4 255.255.255.255
```

In the above example, the command defines the prefix-set named acl-prefix-set. The first element specifies to match all routes having 10 in first octet and 1 in third octet. The second element matches all routes having prefix as 10.2.2.2 (that is, matches all conditions). The third element matches all routes having odd numbers in the last octets and the fourth element matches all routes with any prefix.
prepend as-path

To prepend the AS path with additional autonomous system numbers, use the `prepend as-path` command in route-policy configuration mode.

```
prepend as-path {as-numberparameter|most-recent} [{numberparameter}]
```

**Syntax Description**
- `as-number` Autonomous system number to prepend to the path.
  - Range for 2-byte Autonomous system numbers (ASNs) is 1 to 65535.
  - Range for 4-byte Autonomous system numbers (ASNs) in asplain format is 1 to 4294967295.
  - Range for 4-byte Autonomous system numbers (ASNs) in asdot format is 1.0 to 65535.65535.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”
- `most-recent` Specifies that the most recent autonomous system number should be prepended.
- `number` (Optional) Number of times the autonomous system number should be prepended. Range is 1 to 63.

**Command Default**
The default `number` is 1.

**Command Modes**
Route-policy configuration

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Use the `prepend as-path` command to prepend the AS path with additional autonomous system numbers.

**Note**
The `prepend as-path` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

This command can take one or two arguments. The first argument (either a number or parameter) is the autonomous system number to prepend to the path. The optional second argument (either a number or parameter) is the number of times the autonomous system number should be prepended.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
The following example shows how to prepend the autonomous system number 666.1 to the AS path three times:
RP/0/RP0/CPU0:router(config-rpl)# prepend as-path 666.1 3

The following example shows how to prepend the autonomous system number 666.0 to the AS path one time:

RP/0/RP0/CPU0:router(config-rpl)# prepend as-path 666.0 1
protocol

To check the protocol that installs the route, use the `protocol` command in route-policy configuration mode.

```
protocol  {in(protocol-set)|is(protocol-name)}
```

**Syntax Description**

- `in (protocol-set)` Specifies a member of a set. The `protocol-set` argument accepts the following keywords within parentheses:
  - `bgp` — Border Gateway Protocol (BGP)
  - `connected` — Connected routes
  - `isis` — ISO Intermediate System-to-Intermediate System (IS-IS)
  - `ospf` — Open Shortest Path First (OSPF)
  - `ospfv3` — Open Shortest Path First version 3 (OSPFv3)
  - `rip` — Routing Information Protocol (RIP)
  - `static` — Static routes

Keywords must be separated by a comma.

- `is protocol-name` Specifies a single protocol name, and accepted keywords are similar to the `protocol-set` argument.

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `protocol` command as a conditional expression within an if statement to specify a protocol to install a route.

Use the `in` keyword to determine if a protocol listed in the `protocol-set` is the originator of the route being filtered.

Use the `is` keyword to determine if `protocol-name` is an exact match.

**Note**

For a list of all conditional expressions available within an if statement, see the `if` command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>
The following example shows how to use the `protocol` command as a conditional expression within if statements:

```
RP/0/RP0/CPU0:router(config)# route-policy rip1
RP/0/RP0/CPU0:router(config-rpl)# if protocol in (connected, static) then
  RP/0/RP0/CPU0:router(config-rpl-if)# add rip-metric 2
  RP/0/RP0/CPU0:router(config-rpl-if)# elseif protocol is bgp 1 then
    RP/0/RP0/CPU0:router(config-rpl-elseif)# add rip-metric 3
    RP/0/RP0/CPU0:router(config-rpl-elseif)# elseif protocol is ospf 2 then
      RP/0/RP0/CPU0:router(config-rpl-elseif)# add rip-metric 4
      RP/0/RP0/CPU0:router(config-rpl-elseif)# else
    RP/0/RP0/CPU0:router(config-rpl-elseif)# add rip-metric 5
    RP/0/RP0/CPU0:router(config-rpl-elseif)# endif
  RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config)# router rip
RP/0/RP0/CPU0:router(config-rip)# interface tenGigE 0/1/0/1
RP/0/RP0/CPU0:router(config-rip-if)# route-policy rip1 out
```
**rd in**

To compare the route distinguisher (RD) associated with the route to RDs contained in either a named or an inline RD set, use the **rd in** command in route-policy configuration mode.

```
rd in \{rd-set-name\#inline-rd-set\#parameter\}
```

**Syntax Description**

- **rd-set-name**: Name of an RD set.
- **inline-rd-set**: Inline RD set. The inline RD set must be enclosed in parentheses.
- **parameter**: Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **rd in** command as a conditional expression within an **if** statement to match a destination entry in a named prefix set or inline prefix set.

**Note**

For a list of all conditional expressions available within an **if** statement, see the **if** command.

This command takes either a named RD set or an inline RD set value as an argument. The condition returns true if the destination entry matches any entry in the RD set or inline RD set. An attempt to match an RD using an RD set that is defined but contains no elements returns false.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>read, write</td>
<td>route-policy</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows the **rd in** command with an inline RD set value as an argument:

```
RP/0/RP0/CPU0:router(config)# route-policy
RP/0/RP0/CPU0:router(config-rpl)# if rd in (128.1.0.0/16:100) then
RP/0/RP0/CPU0:router(config-rpl-if)# pass
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**rd-set**

To define a route distinguisher (RD) set and enter RD configuration mode, use the `rd-set` command in XR Config mode.

```
rd-set name
no rd-set name
```

**Syntax Description**

`name` Name of an RD community set.

**Command Default**

No default behavior or values

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `rd-set` command to create a set with RD elements and enter RD configuration mode. An RD set is a 64-bit value prepended to an IPv4 address to create a globally unique Border Gateway Protocol (BGP) VPN IPv4 address.

**Note**

For `m`, the mask length is supported.

You can define RD values with the following commands:

- `a.b.c.d/m:*`—BGP VPN RD in IPv4 format with a wildcard character. For example, `10.0.0.2/24.0:*`.
- `a.b.c.d/m:n`—BGP VPN RD in IPv4 format with a mask. For example, `10.0.0.2/24:666`.
- `a.b.c.d:*`—BGP VPN RD in IPv4 format with a wildcard character. For example, `10.0.0.2:*`.
- `a.b.c.d:n`—BGP VPN RD in IPv4 format. For example, `10.0.0.2:666`.
- `asn:*`—BGP VPN RD in ASN format with a wildcard character. For example, `10002:*`.
- `asn:n`—BGP VPN RD in ASN format. For example, `10002:666`.
- `x.y:*`—BGP VPN RD in 4-byte ASN format with a wildcard character. For example, `10002.101:*`.
- `x.y:n`—BGP VPN RD in 4-byte ASN format. For example, `10002.101:666`.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to create an RD set called `my_rd_set`:

```
RP/0/RP0/CPU0:router(config)# rd-set my_rd_set
RP/0/RP0/CPU0:router(config-rd)# 172.16.0.0/16:*,
RP/0/RP0/CPU0:router(config-rd)# 172.17.0.0/16:100,
```
rd-set

RP/0/RP0/CPU0:router(config-rd)# 192:*,
RP/0/RP0/CPU0:router(config-rd)# 192:100
RP/0/RP0/CPU0:router(config-rd)# end-set
replace as-path

To replace a sequence of AS numbers or private AS numbers in the AS path with the configured local AS number, use the **replace as-path** command in route-policy configuration mode.

```
replace as-path \[\[as-number-list parameter\]\|private-as\]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>as-number-list</strong></td>
<td>(Optional) Sequence of AS numbers to replace. The sequence must be enclosed in single quotes (<code>'</code>). You can use 2-byte or 4-byte AS numbers.</td>
</tr>
<tr>
<td>• The 2-byte value is entered as a 16-bit unsigned decimal value. The range is 0 to 65535.</td>
<td></td>
</tr>
<tr>
<td>• The 4-byte value is entered as two 16-bit unsigned decimal values separated by a period. The range is 1.0 to 65535.65535.</td>
<td></td>
</tr>
<tr>
<td><strong>parameter</strong></td>
<td>(Optional) Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
<tr>
<td><strong>private-as</strong></td>
<td>Matches within the BGP private AS range. Range is from 64512 to 65534.</td>
</tr>
</tbody>
</table>

| Command Default | None. |
| Command Modes | Route-policy configuration |

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **replace as-path** command to replace a sequence of AS numbers or private AS numbers in the AS path with the local AS numbers. For example, if the AS path is ‘67 65534 100 65533 5 78 89 90’ and the local AS number is 900, then:

```
replace as-path '5 78'
```

replaces ‘5 78’ in the AS path with 900 (from the local AS), and the new path would be ‘67 65534 100 65533 900 89 90’.

Consider following statement:

```
replace as-path private-as
```

Because 65534 and 65533 are within the private AS range, they are replaced with 900. The path is ‘67 900 100 900 5 78 89 90’. The length of the path remains the same.

The **replace as-path** command can be used as an action statement within an if statement. For a list of all action statements available within an if statement, see the **if** command.

**Caution**

The **replace as-path** command changes the AS path content which can lead to routing loops.
Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples

The following examples show how to use the `replace as-path` command to replace AS numbers in the AS path:

```
RP/0/RP0/CPU0:router(config)# route-policy drop-as-1234
RP/0/RP0/CPU0:router(config-rpl)# replace as-path '90 78 45 $asnum'
RP/0/RP0/CPU0:router(config-rpl)# replace as-path private-as
RP/0/RP0/CPU0:router(config-rpl)# replace as-path '9.9 7.89 14.15 $asnum'
RP/0/RP0/CPU0:router(config-rpl)# replace as-path '9 89 14.15 $asnum'
```
remove as-path private-as

To remove BGP private AS numbers from as-path structure used by BGP, use the remove as-path private-as command under route policy configuration mode.

remove as-path private-as [ entire-aspath]

Syntax Description

| entire-aspath | (Optional) Removes the entire private autonomous system numbers from an autonomous system path only if all the autonomous systems in the path are private. |

Command Default
No default behavior or values

Command Modes
Route-policy configuration

Command History
Release 6.0  This command was introduced.

Usage Guidelines
No specific guidelines impact the use of this command.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples

This example shows how to remove BGP private AS numbers from as-path structure:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# route-policy rm_private_as
RP/0/RP0/CPU0:router(config-rpl)# remove as-path private-as entire-aspath
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**rib-has-route**

To check if a route listed in the prefix set exists in the Routing Information Base (RIB), use the **rib-has-route** command in route-policy configuration mode.

**Syntax Description**

- **prefix-set-name**  Name of a prefix set.
- **inline-prefix-set**  Inline prefix set. The inline prefix set must be enclosed in parentheses.
- **parameter**  Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If routes are active, then they are advertised. Routes are considered active if they are already installed in the Routing Information Base (RIB).

The prefix sets used in the **rib-has-route** command contain two match specifications. The first is where an exact route match is requested (for example, 10.10.0.0/16 will match exactly one route) and the second is where a route match or any more-specific route match is allowed (for example, 10.10.0.0/16 le 32 will match the 10.10.0.0/16 route and any longer prefix).

Use the **rib-has-route** command as a conditional expression within an **if** statement to check if there is an active route with a specific prefix contained in the RIB. If the statement reveals an active route that meets that criteria, additional actions are executed.

For a list of all conditional expressions available within an **if** statement, see the **if** command.

**Examples**

In the following example, an **if** statement is used to learn if a route contained in a prefix set 10.10.0.0/16 is in the RIB:

```plaintext
RP/0/RP0/CPU0:router(config-rpl)# if rib-has-route in (10.10.0.0/16 ge 16) then
RP/0/RP0/CPU0:router(config-rpl-if)# pass
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
route-has-label

To check if there is a Multiprotocol Label Switching (MPLS) label in a route during redistribution, use the `route-has-label` command in route-policy configuration mode.

**Syntax Description**
This command has no arguments or keywords.

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**
- **Release 6.0**  This command was introduced.

**Usage Guidelines**
Use the `route-has-label` command as a conditional expression within an `if` statement to check if there is an MPLS label in a route during redistribution.

For a list of all conditional expressions available within an `if` statement, see the `if` command.

**Task ID**
Task ID  Operations
route-policy  read, write

**Examples**
In the following example, an `if` statement learns if an MPLS label is present in a route:

```
RP/0/RP0/CPU0:router(config-rpl)# if route-has-label then
RP/0/RP0/CPU0:router(config-rpl-if)# pass
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
```
route-policy (RPL)

To define a route policy and enter route-policy configuration mode, use the **route-policy** command in XR Config mode. To remove a policy definition, use the **no** form of this command.

```
route-policy name [(parameter1, parameter2, . . . , parameterN)]
no route-policy name (parameter1, parameter2, . . . , parameterN)
```

**Syntax Description**

- **name** Name of a route policy.
- **parameter** (Optional) Parameter name. The parameter name must be preceded with a “$.” The **parameters** must be enclosed in parenthesis “()”.

**Command Default**

No default behavior or values

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **route-policy** command to define a route policy and enter route-policy configuration mode.

Policy definitions create named bundles of policy statements. A policy definition consists of the **route-policy** command followed by a name, a group of policy statements, and the **end-policy** command.

The policy name serves as a handle for binding the policy to protocols.

**Examples**

The following example shows a simple policy named drop-everything that drops any route it encounters:

```
RP/0/RP0/CPU0:router(config)# route-policy drop-everything
RP/0/RP0/CPU0:router(config-rpl)# drop
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```

Policies may also refer to other policies such that common blocks of policy can be reused. This reference to other policies is accomplished by using the **apply** command. The following is a simple example:

```
RP/0/RP0/CPU0:router(config)# route-policy drop-as-1234
RP/0/RP0/CPU0:router(config-rpl)# if as-path passes-through '1234' then
RP/0/RP0/CPU0:router(config-rpl-if)# apply check-communities
```
The `apply` command indicates that the policy check-communities should be executed if the route under consideration passed through autonomous system 1234 before it was received. If so, the communities of the route are checked, and based on the findings the route may be accepted unmodified, accepted with changes, or dropped.
route-type is

To match route types when redistribution is being performed into Border Gateway Protocol (BGP), Open Shortest Path First (OSPF), or Integrated Intermediate System-to-Intermediate System (IS-IS), use the **route-type is** command in route-policy configuration mode.

**route-type is** \{local|interarea|internal|type-1|type-2|level-1|level-2\}\{parameter\}

---

**Syntax Description**

- **local**: Uses a local value to match locally generated BGP routes.
- **interarea**: Uses an interarea value to match IS-IS interarea routes.
- **internal**: Uses an internal value to match OSPF intra- and interarea routes.
- **type-1**: Uses a Type 1 value to match Type 1 OSPF routes.
- **type-2**: Uses a Type 2 value to match Type 2 OSPF routes.
- **level-1**: Uses a Level 1 value to match Level 1 IS-IS routes.
- **level-2**: Uses a Level 2 value to match Level 2 IS-IS routes.

**parameter**: Parameter name. The parameter name must be preceded with a “$.”

---

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

Release 6.0  This command was introduced.

**Usage Guidelines**

Use the **route-type is** command as a conditional expression within an **if** statement to compare route types when redistribution is being performed into BGP, OSPF, or IS-IS.

**Note**

For a list of all conditional expressions available within an **if** statement, see the **if** command.

The valid keywords are **local**, **internal**, **interarea**, **type-1**, **type-2**, **level-1**, and **level-2**. A parameterized value that fills in one of these values may also be used. The **local** value is used to match locally generated BGP routes. The internal value is used to match OSPF intra- and interarea routes. The **type-1** and **type-2** values are used to match Type 1 and Type 2 OSPF external routes. The **level-1**, **level-2**, and **interarea** values are used to match IS-IS routes of those respective types.

Because the route type is a matching operator, it appears in conditional clauses of **if** and **then** statements.
In the following example, non-local routes are dropped:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_A
RP/0/RP0/CPU0:router(config-rpl)# if route-type is local then
  RP/0/RP0/CPU0:router(config-rpl-if)# pass
  RP/0/RP0/CPU0:router(config-rpl-if)# else
  RP/0/RP0/CPU0:router(config-rpl-else)# drop
  RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**rpl editor**

To set the default routing policy language (RPL) editor, use the `rpl editor` command in XR Config mode.

```
rpl editor {nano|emacs|vim}
```

**Syntax Description**
- `nano` Sets the default RPL editor to GNU nano.
- `emacs` Sets the default RPL editor to EMACS.
- `vim` Sets the default RPL editor to VIM.

**Command Default**
The Nano editor is the default.

**Command Modes**
XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>read,</td>
<td>write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, the default RPL editor is set to Nano:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rpl editor nano
```

In the following example, the default RPL editor is set to EMACS:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rpl editor emacs
```

In the following example, the default RPL editor is set to VIM:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rpl editor vim
```
rpl maximum

To configure system limits on the routing policy subsystem, use the `rpl maximum` command in XR Config mode.

```
rpl maximum {lines|policies} number
```

**Syntax Description**
- `lines number` Configures the number of lines of configuration limit. Range is from 1 to 131072.
- `policies number` Configures the number of policies limit. Range is from 1 to 5000.

**Command Default**
- `lines number : 65536`
- `policies numbers : 3500`

**Command Modes**
- XR Config mode

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Use the `rpl maximum` command to configure system limits on the routing policy subsystem. As such, `rpl maximum` configuration lines do not appear as statements within a routing policy. This command places resource limits on the routing policy subsystem. Use the `rpl maximum` command to configure the maximum number of lines of configuration and number of policies.

The number of lines of configuration includes the beginning and ending statements, for example, `route-policy` and `end-policy`. Each line of configuration for sets is also counted.

A line of configuration is counted only once; it is not counted each time it is used. Similarly, any multiple use of policy in an apply statement counts only as one policy.

A user can change the default values for lines and policies but cannot exceed the maximum value, nor can the value for lines and policies be configured lower than the number of lines or policies that are currently configured.

**Task ID**
- `read, write`

**Examples**
In the following example, the maximum number of RPL system limits are modified:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rpl maximum lines 50
RP/0/RP0/CPU0:router(config)# rpl maximum policies 6
```
**rpl set-exit-as-abort**

To change the default exit behavior under RPL configuration mode to abort from the RPL configuration mode without saving the configuration, use the *rpl set-exit-as-abort* command in XR Config mode.

**Syntax Description**

This command has no keywords or arguments.

**Command Modes**

XR Config mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

The default *exit* command acts as *end-policy, end-set, or end-if*. If the *exit* command is executed under route policy configuration mode, the changes are applied and configuration is updated. This destructs the existing policy. The *rpl set-exit-as-abort* command allows to overwrite the default behavior of the *exit* command under the route policy configuration mode.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

This example shows how change the default exit behavior:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# rpl set-exit-as-abort
```
set administrative-distance

To set a route with lower administrative distance such that it is preferred to a route with higher administrative distance, use the set administrative-distance command in route policy configuration mode.

```
set administrative-distance [{number | parameter}]
```

**Syntax Description**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>number</td>
<td>Value assigned to a 8-bit unsigned integer. Range is from 1 to 255.</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter name. The parameter name must be preceded with a &quot;$&quot;.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

This example shows how to set a route with an administrative value such that it is preferred to a route with higher administrative distance.

```bash
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# route-policy sample
RP/0/RP0/CPU0:router(config-rpl)# set administrative-distance 34
RP/0/RP0/CPU0:router(config-rpl)# end-policy
RP/0/RP0/CPU0:router(config-rpl)# exit
RP/0/RP0/CPU0:router(config)# route bgp 100
RP/0/RP0/CPU0:router(config-bgp)# address family ipv4 unicast
RP/0/RP0/CPU0:router(config-bgp-af)# table-policy sample
RP/0/RP0/CPU0:router(config-bgp-af)# exit
RP/0/RP0/CPU0:router(config-bgp)# exit
RP/0/RP0/CPU0:router(config)# end
```
set aigp-metric

To set originating prefixes with accumulated interior gateway protocol (AiGP) attribute information, use the `set aigp-metric` command in route-policy configuration mode.

```
set aigp-metric {igp-cost value}
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>igp-cost</td>
<td>Specifies the internal routing protocol cost.</td>
</tr>
<tr>
<td>value</td>
<td>Specifies the aigp-metric value. 32-bit decimal number. Range is 0-4294967295.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>No default behavior or values</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command Modes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route-policy configuration</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Command History</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
</tr>
<tr>
<td>Modification: This command was introduced.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Usage Guidelines</th>
</tr>
</thead>
<tbody>
<tr>
<td>No specific guidelines impact the use of this command.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation</td>
</tr>
<tr>
<td>route-policy</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>The following example shows how to set the aigp-metric as the igp cost for route-policy aigp_policy:</td>
</tr>
</tbody>
</table>

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# route-policy aigp_policy
RP/0/RP0/CPU0:router(config-rpl)# set aigp-metric igp-cost
```
**set community**

To set the Border Gateway Protocol (BGP) community attributes in a route, use the `set community` command in route-policy configuration mode.

```
set community {community-set-name inline-community-set parameter} [additive]
```

**Syntax Description**

- `community-set-name` Community set name.
- `inline-community-set` Inline community set. The inline community set must be enclosed in parentheses.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”
- `additive` (Optional) Adds communities to communities in the route.

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set community` command to set the BGP community attribute.

The `set community` command can be used as an action statement within an if statement. For a list of all action statements available within an if statement, see the `if` command.

Communities are 32-bit values carried in BGP routes. Each route may have zero or more communities in an unordered list.

Use this command to replace the communities in a route or add to them using the optional `additive` keyword.

As with the other community forms that support inline sets, either or both 16-bit portions of the community can be parameterized. Likewise, the names of the well-known communities `internet` (0:0), `no-advertise` (65535:65281), `no-export` (65535:65282), and `local-AS` (65535:65283) can also be used. In an inline community set, each 16-bit portion can also be specified as the `peeras` to express the AS number of the neighbor from which the route was received. If the neighbor AS employs a 4-byte ASN, the IANA-assigned 16-bit value 23456 (AS_TRANS) is used as `peeras` instead.

Without the `additive` keyword, any existing communities (other than the well-known communities) are removed and replaced with the given communities. The `additive` keyword specifies that all communities already present in the route be maintained and the list of communities be added to them.
The following are incomplete configuration examples using the `set community` command:

```
RP/O/RP0/CPU0:router(config-rpl)# set community (10:24)
RP/O/RP0/CPU0:router(config-rpl)# set community (10:24, $as:24, $as:$tag)
RP/O/RP0/CPU0:router(config-rpl)# set community (10:24, internet) additive
RP/O/RP0/CPU0:router(config-rpl)# set community (10:24, $as:24) additive
RP/O/RP0/CPU0:router(config-rpl)# set community (10:24, peeras:24) additive
```
set core-tree

To set a Multicast Distribution Tree (MDT) type, use the `set core-tree` command in route-policy configuration mode.

```
set core-tree
    [gre-rosen|mldp-inband|mldp-partitioned-mp2mp|mldp-partitioned-p2mp|mldp-rosen|rsvp-te-partitioned-p2mp] parameter
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>gre-rosen</code></td>
<td>Specifies the IP GRE Rosen core MDT type</td>
</tr>
<tr>
<td><code>mldp-inband</code></td>
<td>Specifies the MLDP InBand core MDT type</td>
</tr>
<tr>
<td><code>mldp-partitioned-mp2mp</code></td>
<td>Specifies the MLDP Partitioned MP2MP core MDT type</td>
</tr>
<tr>
<td><code>mldp-partitioned-p2mp</code></td>
<td>Specifies the MLDP Partitioned P2MP core MDT type</td>
</tr>
<tr>
<td><code>mldp-rosen</code></td>
<td>Specifies the MLDP Rosen core MDT type</td>
</tr>
<tr>
<td><code>rsvp-te-partitioned-p2mp</code></td>
<td>Specifies the RSVP TE core core MDT type</td>
</tr>
<tr>
<td><code>parameter</code></td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
</tbody>
</table>

**Command Default**

None

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read,</td>
</tr>
<tr>
<td></td>
<td>write</td>
</tr>
</tbody>
</table>

In this example, the Multicast Distribution Tree type is set to IP GRE Rosen core:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#route-policy policy_mdt_type
RP/0/RP0/CPU0:router(config)#set core-tree gre-rosen
```
**set dampening**

To configure Border Gateway Protocol (BGP) route dampening, use the `set dampening` command in route-policy configuration mode.

```
set dampening {halflife {minutes parameter}|max-suppress {minutes parameter}|reuse {seconds parameter}|suppress {penalty-units parameter}|others default}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>halflife</strong></td>
<td><em>minutes</em> Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
<tr>
<td><strong>max-suppress</strong></td>
<td><em>minutes</em> Specifies the maximum time (in minutes) a route can be suppressed. Range is 1 to 20000. If the half-life value is allowed to default, the maximum suppress time defaults to 60 minutes.</td>
</tr>
<tr>
<td><strong>reuse</strong></td>
<td><em>seconds</em> Unsuppresses a route if the penalty for flapping the route decreases enough to fall below the configured value (in seconds). The process of unsuppressing routes occurs at 10-second increments. Range is 1 to 20000.</td>
</tr>
<tr>
<td><strong>suppress</strong></td>
<td><em>penalty-units</em> Specifies a penalty of 1000 each time a route flaps. When a route penalty exceeds the configured limit, it is suppressed. Range is 1 to 20000.</td>
</tr>
<tr>
<td><strong>others default</strong></td>
<td>If all four keyword values are not specified in the command, then the command must end with <strong>others default</strong>. This designation indicates that any keyword not defined is set to its default.</td>
</tr>
</tbody>
</table>

**Command Default**

- **half-life**: 15 minutes
- **max-suppress**: 60 minutes (four times the half-life)
- **reuse**: 750 seconds
- **suppress**: 2000 penalty units

**Command Modes**

- Route-policy configuration

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

The BGP protocol supports route dampening using an exponential backoff algorithm. The algorithm is controlled by setting the four supported BGP values: half-life, max-suppress, reuse, and suppress. Use the `set dampening` command to configure BGP route dampening.
The **set dampening** command can be used as an action statement within an **if** statement. For a list of all action statements available within an **if** statement, see the **if** command.

A value for at least one of the four keywords must be set. If the **set dampening** command defines values for three or fewer of the supported keywords, then the configuration must end with the **others default**, which indicates that any keyword value not defined in the command is set to its default value.

The keywords may appear in the command in any order.

### Examples

In the following examples, the half-life is set to 20 minutes and the maximum suppress time is set to 90 minutes. Each command must end with **others default** because three or fewer keywords are defined.

```
RP/0/RP0/CPU0:router(config-rpl)# set dampening halflife 20 others default
RP/0/RP0/CPU0:router(config-rpl)# set dampening max-suppress 90 others default
```

In this example, all four keywords are defined, which means the command does not use **others default**.

```
RP/0/RP0/CPU0:router(config-rpl)# set dampening halflife 15 max-suppress 60 reuse 750 suppress 2000
```

The following command is invalid because it is missing **others default**.

```
RP/0/RP0/CPU0:router(config-rpl)# set dampening reuse 700
```

In the following example, the parameters are used.

```
RP/0/RP0/CPU0:router(config-rpl)# set dampening halflife $p1 suppress $p4 reuse $p3 max-suppress $p2
```
set extcommunity cost

To set the Border Gateway Protocol (BGP) cost extended community attributes, use the `set extcommunity cost` command in route-policy configuration mode.

**Syntax**

```
set extcommunity cost \{cost-extcommunity-set-name\} \{cost-inline-extcommunity-set\} \{parameter\} \{additive\}
```

**Syntax Description**

- `cost-extcommunity-set-name`: Cost extended community set name.
- `cost-inline-extcommunity-set`: Inline cost extended community set. The inline cost extended community set must be enclosed in parentheses.
- `parameter`: Parameter name. The parameter name must be preceded with a “$.”
- `additive`: (Optional) Adds extended communities for cost to extended communities in the route.

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set extcommunity cost` command to either replace the extended communities on the route or add to them using the optional `additive` keyword. Cost community is an extended community used to tie break the best path selection process in BGP so as to have a localized custom decision for packet forwarding. The extended community format defines generic points of insertion (POI) that influence the decision at different points of the best path algorithm.

As with the other extended community forms that support inline sets, either or both portions of the community can be parameterized. Similarly to regular communities, the `additive` keyword can be used to signify adding these extended communities to those that are already present, as opposed to replacing them. Without the `additive` keyword, any existing extended communities for cost (other than the well-known communities) are removed and replaced with the given communities. The `additive` keyword specifies that all extended communities for cost already present in the route be maintained and the set of extended communities be added to them. Well-known communities include internet, local-AS, no-advertise, and no-export.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>
Examples

The following are incomplete configuration examples using the `set extcommunity cost` command:

RP/0/RP0/CPU0:router(config-rpl)# `set extcommunity cost (IGP:10:20)`
RP/0/RP0/CPU0:router(config-rpl)# `set extcommunity cost (Pre-Bestpath:33:44)`
RP/0/RP0/CPU0:router(config-rpl)# `set extcommunity cost (IGP:11:21)`
set extcommunity rt

To set the Border Gateway Protocol (BGP) route target (RT) extended community attributes, use the `set extcommunity rt` command in route-policy configuration mode.

```
set extcommunity rt \{rt-extcommunity-set-namert-inline-extcommunity-setparameter\} additive
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rt-extcommunity-set-name</td>
<td>Route target extended community set name.</td>
</tr>
<tr>
<td>rt-inline-extcommunity-set</td>
<td>Inline route target extended community set. The inline route target extended community set must be enclosed in parentheses.</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
<tr>
<td>additive</td>
<td>(Optional) Adds extended communities for an RT to extended communities in the route.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set extcommunity rt` command to either replace the extended communities on the route or add to them using the optional `additive` keyword.

**Note**

The `set extcommunity rt` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

As with the other extended community forms that support inline sets, either or both portions of the community can be parameterized. Similarly to regular communities, the `additive` keyword can be used to signify adding these extended communities to those that are already present, as opposed to replacing them.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following are incomplete configuration examples using the `set extcommunity rt` command:

```
RP/0/RP0/CPU0:router(config-rpl)# set extcommunity rt (10:24)
RP/0/RP0/CPU0:router(config-rpl)# set extcommunity rt (10:24, $as:24, $as:$tag)
RP/0/RP0/CPU0:router(config-rpl)# set extcommunity rt (10:24, internet) additive
```
Without the **additive** keyword, any existing extended communities for cost (other than the well-known communities) are removed and replaced with the given communities. The **additive** keyword specifies that all extended communities for cost already present in the route be maintained and the list of extended communities be added to them.
set ip-precedence

To set the IP precedence, use the `set ip-precedence` command in route-policy configuration mode.

```
set ip-precedence {number parameter}
```

**Syntax Description**

- **number**: Value of the precedence. The precedence value can be a number from 0 to 7:
  - 7 — network (set packets with network control precedence)
  - 6 — internet (set packets with internetwork control precedence)
  - 5 — critical (set packets with critical precedence)
  - 4 — flash-override (set packets with flash override precedence)
  - 3 — flash (set packets with flash precedence)
  - 2 — immediate (set packets with immediate precedence)
  - 1 — priority (set packets with priority precedence)
  - 0 — routine (set packets with routine precedence)

- **parameter**: Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set ip-precedence` command to set the IP precedence to classify packets. This command is supported at the BGP table-policy attachpoint. Prefixes are marked for subsequent processing in the forwarding plane. After QoS Policy Propagation through Border Gateway Protocol (BGP) (QPPB) is enabled on an interface, corresponding traffic shaping and policing is completed using packet classification based on the IP precedence or QoS group ID. See Modular QoS Configuration Guide for Cisco NCS 5000 Series Routers for information on QPPB.

**Task ID**

- `route-policy` read, write

**Examples**

This example shows how use `set ip-precedence` command:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# set ip-precedence 3
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
set isis-metric

To set the Intermediate System-to-Intermediate System (IS-IS) metric attribute value, use the set isis metric command in route-policy configuration mode.

```
set isis-metric \{numberparameter\}
```

**Syntax Description**

- `number` 24-bit integer number. Range is from 0 to 16777215.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the set isis-metric command to set the IS-IS metric attribute value for routes that are redistributed into IS-IS.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, the IS-IS metric attribute value is set to 1000:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# set isis-metric 1000
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**set label**

To set the Border Gateway Protocol (BGP) label attribute value, use the `set label` command in route-policy configuration mode.

```plaintext
set label {explicit-null|implicit-null}parameter
```

**Syntax Description**

- **explicit-null** Sets the label to the well-known explicit value of 0.
- **implicit-null** Sets the label to the well-known implicit value of 3.
- **parameter** Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

Use the `set label` command in a route policy at the allocate label attachpoint to set the label to explicit-null or implicit-null based on deployment preference. During inter-AS operation, the ASBR sends some of its own loopbacks to other its peers and labels them either implicit null or explicit null.

**Examples**

The following example shows how to set the labels:

```plaintext
RP/0/RP0/CPU0:router(config)# route-policy labelpolicy
RP/0/RP0/CPU0:router(config-rpl)# if destination in (206.141.1.0/24) then
RP/0/RP0/CPU0:router(config-rpl)# set label explicit-null
RP/0/RP0/CPU0:router(config-rpl)# elseif destination in (206.141.3.0/24) then
RP/0/RP0/CPU0:router(config-rpl)# drop
RP/0/RP0/CPU0:router(config-rpl)# elseif destination in (206.141.4.0/24) then
RP/0/RP0/CPU0:router(config-rpl)# set label explicit-null
RP/0/RP0/CPU0:router(config-rpl)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
set label-mode

To set the type of Border Gateway Protocol (BGP) label mode, use the `set label-mode` command in route-policy configuration mode. This command does not have a `no` form.

```
set label-mode {per-ce | per-vrf | per-prefix}
```

**Syntax Description**

- **per-ce**  Specifies that the same label is used for all routes advertised from a unique customer edge (CE) peer or router.
- **per-vrf** Specifies that the same label is used for all routes advertised from a unique VRF.
- **per-prefix** Specifies that the same label is used for all routes advertised from a unique prefix.

**Command Default**

Per-prefix label mode.

If a policy attached at label-mode attachpoint evaluates to pass and a label mode is not explicitly set, `per-prefix` is used as a default label mode.

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set label-mode` command in a route policy at the label-mode attachpoint to set the type of label mode to `per-ce` or `per-vrf` or `per-prefix`, based on deployment preference.

Per-vrf label mode is not supported for Carrier Supporting Carrier (CSC) network with internal and external BGP multipath setup.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example shows how to set the type of label-mode to `per-ce`:

```
RP/0/RP0/CPU0:router(config)# route-policy set_label_mode
RP/0/RP0/CPU0:router(config-rpl)# set label-mode per-ce
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```

This example shows how to set the type of label-mode to `per-vrf`:

```
RP/0/RP0/CPU0:router(config)# route-policy set_label_mode
RP/0/RP0/CPU0:router(config-rpl)# set label-mode per-vrf
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```

This example shows how to set the type of label-mode to `per-prefix`:
RP/0/RP0/CPU0:router(config)# route-policy set_label_mode
RP/0/RP0/CPU0:router(config-rpl)# set label-mode per-prefix
RP/0/RP0/CPU0:router(config-rpl)# end-policy
set level

To configure the Intermediate System-to-Intermediate System (IS-IS) link-state packet (LSP) level advertised to redistributed routes, use the set level command in route-policy configuration mode.

```
set level {level-1|level-2|level-1-2}parameter
```

**Syntax Description**

- **level-1**: Specifies that redistributed routes are advertised in the Level 1 LSP of the router.
- **level-2**: Specifies that redistributed routes are advertised in the Level 2 LSP of the router.
- **level-1-2**: Specifies that redistributed routes are advertised in Level 1 and Level 2 LSPs of the router.
- **parameter**: Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the IS-IS set level command to configure the LSP level advertised to redistributed routes.

**Note**

The set level command can be used as an action statement within an if statement. For a list of all action statements available within an if statement, see the if command.

This command supports parameterization of the level keyword.

**Task ID**

Task ID | Operations
--------|-------------
read, write

**Examples**

In the following example, the level is set to Level 2:

```
RP/0/RP0/CPU0:router(config)# route-policy bgp_isis_redist
RP/0/RP0/CPU0:router(config-rpl)# if destination in (172.2.0.0/16 ge 16) then
RP/0/RP0/CPU0:router(config-rpl)# set level level-2
RP/0/RP0/CPU0:router(config-rpl)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
set local-preference

To set the Border Gateway Protocol (BGP) local preference attribute in a route, use the `set local-preference` command in route-policy configuration mode.

```
set local-preference {number parameter}
```

**Syntax Description**
- `number` Value assigned to a 32-bit unsigned integer. Range is 0 to 4294967295.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
Default value is 100.

**Command Modes**
Route-policy configuration

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Use the `set local-preference` command to specify a preference value for the autonomous system path. Local preference is a nontransitive (does not cross autonomous system boundaries) attribute and is the second metric considered in the BGP best path calculation (the highest local preference is chosen). Weight is the first metric evaluated for best path, but it is local to the router and propagates only to iBGP peers. See the Implementing BGP module of the Routing Configuration Guide for Cisco NCS 5000 Series Routers for information on the BGP best path calculation.

**Note**
The `set local-preference` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

The local preference is a 32-bit unsigned integer.

**Examples**
In the following example, the local preference value is set to 10:

```
RP/0/RP0/CPU0:router(config-rpl)# set local-preference 10
```
**set med**

To set the Border Gateway Protocol (BGP) Multi Exit Discriminator (MED) attribute, use the `set med` command in route-policy configuration mode.

```
set med {numberparameter|igp-cost|{+|{-}numberparameter}}|max-reachable}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>number</code></td>
<td>Value assigned to a 32-bit unsigned integer. Range is 0 to 4294967295.</td>
</tr>
<tr>
<td><code>parameter</code></td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
<tr>
<td><code>igp-cost</code></td>
<td>Sets the MED value to the cost for the Interior Gateway Protocol (IGP) route to resolve the next-hop of the BGP route.</td>
</tr>
<tr>
<td>`+</td>
<td>-`</td>
</tr>
<tr>
<td><code>max-reachable</code></td>
<td>Sets the MED value to the maximum possible value of 4294967295.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set med` command to set the MED value, which is a 32-bit unsigned integer.

The `set med` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

This command can take the following as argument values: an integer, a parameter, the `igp-cost` keyword, or a mathematical operator (either plus or minus) followed by an integer or a parameter. Setting the MED to the IGP cost is supported on outbound BGP policies only. The MED cannot be set to the IGP cost in policies applied to other BGP attachment points.

The `max-reachable` keyword sets the MED to the maximum value while leaving the route reachable.

The plus or minus variants allow the user to set the MED to the MED plus or minus a static offset. The variants that allow a user to add or subtract offsets to the MED value are also range checked for underflow or overflow. If the value underflows as a result of subtraction, then the MED value is set to zero. If the value overflows, the value is set to 4294967295, which is the maximum value for MED. when MED is set to 4294967295, the route is unreachable.
The following two examples show how to set the MED to a value that is either specified directly (using the integer 156) or passed to the policy as a parameter:

```plaintext
RP/0/RP0/CPU0:router(config-rpl)# set med 156
RP/0/RP0/CPU0:router(config-rpl)# set med $med_param
```

The following example shows how to instruct BGP to automatically set the MED value to the cost of the IGP route that resolves the next-hop of the BGP route:

```plaintext
RP/0/RP0/CPU0:router(config-rpl)# set med igp-cost
```
set metric-type (IS-IS)

To configure the integrated Intermediate System-to-Intermediate System (IS-IS) metric type, use the `set metric-type` command in route-policy configuration mode.

```
set metric-type {internal|external|rib-metric-as-internal| rib-metric-as-external}parameter`
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>internal</code></td>
</tr>
<tr>
<td><code>external</code></td>
</tr>
<tr>
<td><code>rib-metric-as-internal</code></td>
</tr>
<tr>
<td><code>rib-metric-as-external</code></td>
</tr>
<tr>
<td><code>parameter</code></td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

User the IS-IS `set metric-type` command to control whether IS-IS treats the metric as an internal or external metric.

Use the `rib-metric-as-external` and `rib-metric-as-internal` keywords to preserve RIB metrics when redistributing routes from another ISIS router instance or another protocol.

**Note**

The `set metric-type` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

This command does not support parameterization.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>
In the following example, the IS-IS metric type is set to internal:

```
RP/0/RP0/CPU0:router(config-rpl)# set metric-type internal
```
set metric-type (OSPF)

To control how Open Shortest Path First (OSPF) computes the cost for a route, use the set metric-type command in route-policy configuration mode.

```
set metric-type {type-1|type-2\param\}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type-1</td>
<td>Uses the cost set on the route plus the topology-related costs in the calculation for Type 1 metrics.</td>
</tr>
<tr>
<td>type-2</td>
<td>Uses only the cost set on the route in the calculation for Type 2 metrics.</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the OSPF set metric-type command to control whether OSPF treats the cost as a Type 1 or Type 2 metric.

The set metric-type command can be used as an action statement within an if statement. For a list of all action statements available within an if statement, see the if command.

The value of Type 1 or Type 2 controls how OSPF computes the cost for this route. For Type 2 metrics, only the cost set on the route is used. For Type 1 metrics, the cost set on the route plus the topology-related costs are used in the calculation.

This command does not support parameterization.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, the OSPF metric type is set to Type 1:

```
RP/0/RP0/CPU0:router(config-rpl)# set metric-type type-1
```
set next-hop

To replace the next-hop associated with a given route, use the `set next-hop` command in route-policy configuration mode.

```
set next-hop {ipv4-address [destination-vrf] ipv6-address [destination-vrf] | discard parameter | peer-address | self}
```

**Syntax Description**

| `ipv4-address` | Valid IPv4 address. |
| `ipv6-address` | Valid IPv6 address. |
| `discard` | Sets next-hop as Null0 interface. |
| `destination-vrf` | (Optional) Specifies that the next-hop of the route should be resolved in destination VRF context. This keyword is available when an IPv4 or IPv6 address or parameter is used. |
| `peer-address` | Sets the next-hop to the IP address of the remote Border Gateway Protocol (BGP) peer. |
| `parameter` | Parameter name. The parameter name must be preceded with a “$.” |
| `self` | Sets itself as the next-hop. |

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set next-hop` command to replace the next-hop associated with a specific address.

**Note**

The `set next-hop` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

Use the `set next-hop peer-address` command to set the next-hop to the address of the BGP neighbor, where this policy is attached.

The next-hop is a valid IPv4 address entered as a dotted decimal or an IPv6 address entered as a colon-separated hexadecimal.

It is not possible to use this command to set the BGP IPv6 link-local next-hop.

The `destination-vrf` keyword is used mainly in Layer 3 VPN networks when importing routes.

The below address families support the selective setting of ‘next-hop-self’ via the RPL statement ‘set next-hop self’ starting in 4.2.1. Previous to this the setting of next-hop-self via an RPL was for all prefixes in the address family or none of the prefixes.
The set next-hop discard configuration is used in the neighbor inbound policy. When this config is applied to a path, the primary next-hop is still be associated with the actual path but the RIB is updated with next-hop set to Null0. Even if the primary received nexthop is unreachable, the Remotely Triggered Blackhole (RTBH) path will be considered reachable and will be a candidate in the bestpath selection process. The RTBH path is readvertised to other peers with either the received next-hop or nexthop-self based on normal BGP advertisement rules.

### Examples

In the following example, the next-hop is set to a valid IPv4 address:

```plaintext
RP/0/RP0/CPU0:router(config-rpl)# set next-hop 10.0.0.5
```

In this example, the next-hop is set to a parameter value $nexthop:

```plaintext
RP/0/RP0/CPU0:router(config-rpl)# set next-hop $nexthop
```

In this example, the next-hop is set to a valid IPv4 address with a destination VRF context:

```plaintext
RP/0/RP0/CPU0:router(config-rpl)# set next-hop 10.0.0.5 destination-vrf
```
**set origin**

To change the Border Gateway Protocol (BGP) origin attribute, use the `set origin` command in route-policy configuration mode.

```
set origin {igp|incomplete|egp}parameter
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>igp</td>
<td>Sets the origin type to Interior Gateway Protocol (IGP).</td>
</tr>
<tr>
<td>incomplete</td>
<td>Sets the origin type to incomplete.</td>
</tr>
<tr>
<td>egp</td>
<td>Sets the origin type to Exterior Gateway Protocol (EGP).</td>
</tr>
</tbody>
</table>

**Parameter name.** Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set origin` command to change the origin attribute.

**Note**

The `set origin` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

The origin of a Border Gateway Protocol (BGP) route is `igp`, `egp`, or `incomplete`.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, the origin attribute is set to EGP:

```
RP/0/RP0/CPU0:router(config-rpl)# set origin egp
```
**set ospf-metric**

To set an Open Shortest Path First (OSPF) protocol metric attribute value, use the `set ospf-metric` command in route-policy configuration mode.

```
set ospf-metric {number|parameter}
```

**Syntax Description**

- `number`: Value assigned to a 24-bit unsigned integer. Range is 0 to 4294967295.
- `parameter`: Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

- **Release**: 6.0  This command was introduced.

**Usage Guidelines**

Use the `set ospf-metric` command to set the metric for routes that are redistributed into OSPF. The OSPF metric operator accepts either an integer value or a parameter.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, the OSPF metric attribute value is set to 1000:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# set ospf-metric 1000
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**set path-selection**

To set path selection criteria and install or advertise the path for the Border Gateway Protocol, use the `set path-selection` command in route-policy configuration mode.

```
set path-selection {backup number|group-best|all|best-path} [install] [multipath-protect] [advertise]
```

### Syntax Description

- **backup** Specifies the BGP backup path.
- **number** Specifies the BGP backup path number. 3 bit decimal number. Range is 0-7.
- **group-best** Specifies the BGP group best path.
- **all** Specifies all BGP paths.
- **best-path** Specifies the BGP best path.
- **install** Installs the path.
- **multipath-protect** Installs and advertises the multipath protect.
- **advertise** Advertises the path.

### Command Default

None

### Command Modes

Route-policy configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

No specific guidelines impact the use of this command.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

### Examples

The following example shows how to set the path selection as **advertise backup path 3** for route-policy **path_selection_plyc**:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# route-policy path_selection_plyc
RP/0/RP0/CPU0:router(config-rpl)# set path-selection backup 3 advertise
```
set qos-group (RPL)

To set the quality of service (QoS) group, use the `set qos-group` command in route-policy configuration mode:

```
set qos-group {numberparameter}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>number</td>
<td>QoS group ID. Range is from 0 to 31.</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set qos-group` command to set the QoS group to classify packets.

This command is supported at the BGP table-policy attachpoint. Prefixes are marked for subsequent processing in the forwarding plane. After QoS Policy Propagation through Border Gateway Protocol (BGP) (QPPB) is enabled on an interface, corresponding traffic shaping and policing is completed using packet classification based on the IP precedence or QoS group ID. See the *Modular QoS Configuration Guide for Cisco NCS 5000 Series Routers* for information on QPPB.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

This example shows how to use `set qos-group` command:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# set qos-group 12
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
set rib-metric

To set the Routing Information Base (RIB) metric attribute value for a table policy, use the `set rib-metric` command in route-policy configuration mode:

```
set rib-metric \{number\parameter\}
```

**Syntax Description**
- *number* Value assigned to a 32-bit unsigned integer. Range is 0 to 4294967295.
- *parameter* Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `set rib-metric` command set the RIB metric attribute value for BGP routes.

Every route in the RIB has a metric associated with it, signifying the cost to reach a specific destination based on link characteristics. The `set rib-metric` command modifies the RIB metric while installing BGP routes into RIB, enabling the upgrading or downgrading of the BGP route installed in RIB.

**Task ID**
- Task ID | Operations
  - route-policy | read, write

**Examples**
In the following example, the RIB metric attribute is set to 1000:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# set rib-metric 1000
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
set rip-metric

To set Routing Information Protocol (RIP) metric attributes, use the **set rip-metric** command in route-policy configuration mode.

```
set rip-metric \{number\parameter\}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*number*</td>
<td>Value assigned to a 4-bit unsigned integer. Range is from 0 to 16.</td>
</tr>
<tr>
<td>*parameter*</td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
</tbody>
</table>

### Command Default

No default behavior or values

### Command Modes

Route-policy configuration

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `set rip-metric` command to set the cost attribute for routes that are redistributed into RIP.

You can use the `add` command to increment the RIP metric value.

### Examples

In the following example, the RIP metric number is adjusted for route policy policy_1:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# set rip-metric 10
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
set rip-tag

To set a route tag attribute for Routing Information Protocol (RIP) routes, use the `set rip-tag` command in route-policy configuration mode.

```
set rip-tag {number|parameter}
```

**Syntax Description**

- `number`: Value assigned to a 16-bit unsigned integer. Range is from 0 to 65535.
- `parameter`: Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set rip-tag` command to set the RIP tag attribute for routes that are redistributed into RIP. The RIP tag operator accepts either an integer value or a parameter.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, the RIP tag is adjusted for route policy policy_1:

```
RP/0/RP0/CPU0:router(config)# route-policy policy_1
RP/0/RP0/CPU0:router(config-rpl)# set rip-tag 1000
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```
**set rpf-topology**

To set reverse-path forwarding (RPF) to any default or nondefault tables for particular sources and groups, use the `set rpf-topology` command in routing policy configuration mode.

```
set rpf-topology [vrf vrf-name] {ipv4|ipv6} {unicast|multicast} parameter topology table-name
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>vrf vrf-name</td>
<td>[Optional] Specifies a VPN routing and forwarding (VRF) instance. Required when configuring extranet topologies</td>
</tr>
<tr>
<td>ipv4</td>
<td>[Optional] Specifies IPv4 address prefixes.</td>
</tr>
<tr>
<td>ipv6</td>
<td>[Optional] Specifies IPv6 address prefixes.</td>
</tr>
<tr>
<td>unicast</td>
<td>Specifies unicast address prefixes.</td>
</tr>
<tr>
<td>multicast</td>
<td>Specifies multicast address prefixes.</td>
</tr>
<tr>
<td>parameter</td>
<td>Parameter name. The parameter name must be preceded with a “$.”</td>
</tr>
<tr>
<td>topology</td>
<td>Specifies the default or non-default topology table for the source or group.</td>
</tr>
<tr>
<td>table-name</td>
<td>Alphanumeric name string.</td>
</tr>
</tbody>
</table>

**Command Default**

Default or current topology setting.

**Command Modes**

Routing policy configuration

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

When using this command for MVPN extranet routing configuration, only the `vrf vrf-name` keyword and argument are required.

When using this command in the context of multitopology routing, all keywords and arguments with the exception of `vrf vrf-name` keyword and argument are required.

**Task ID**

```
Task ID      Operations
multicast    read
```

**Examples**

The following example shows how to execute the `set rpf-topology` command:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# route-policy green
RP/0/RP0/CPU0:router(config-rpl)# set rpf-topology ipv6 multicast topology t12
```
The following example shows the use of `set rpf-topology` command in the context of creating an RPF for a topology table in multiple topologies:

```
route-policy mt4-p1
  if destination in (225.0.0.1, 225.0.0.11) then
    set rpf-topology ipv4 multicast topology t201
  elseif destination in (225.0.0.2, 225.0.0.12) then
    set rpf-topology ipv4 multicast topology t202
  elseif destination in (225.0.0.3, 225.0.0.13) then
    pass
  endif
end-policy
!

route-policy mt4-p3
  if destination in (225.0.0.8) then
    set rpf-topology ipv4 multicast topology t208
  elseif destination in (225.0.0.9) then
    set rpf-topology ipv4 multicast topology t209
  elseif destination in (225.0.0.10) then
    set rpf-topology ipv4 multicast topology t210
  else
    drop
  endif
end-policy
!
```
set spf-priority

To set OSPF Shortest Path First (SPF) priority, use the set spf-priority command in route-policy configuration mode.

```
set spf-priority  {critical|high|medium}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>critical</th>
<th>Sets critical priority for SPF</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>high</td>
<td>Sets high priority for SPF</td>
</tr>
<tr>
<td></td>
<td>medium</td>
<td>Sets medium priority for SPF</td>
</tr>
</tbody>
</table>

**Command Default**

None

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

This example sets SPF priority as critical:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#route-policy policy_spf_priority
RP/0/RP0/CPU0:router(config-rpl)#set spf-priority critical
```
set tag

To set the tag attribute, use the `set tag` command in route-policy configuration mode.

```
set tag {numberparameter}
```

**Syntax Description**

- `number`  Value assigned to a 32-bit unsigned integer. Range is from 0 to 4294967295.
- `parameter`  Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `set tag` command to set the tag attribute.

**Note**

The `set tag` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

Tags are routing-protocol independent 32-bit integers that can be associated with a given route in the Routing Information Base (RIB).

For the Border Gateway Protocol (BGP), the tag attribute can be set only at the table-policy attach point.

**Examples**

In the following example, the tag attribute is set to 10:

```
RP/0/RP0/CPU0:router(config-rpl)# set tag 10
```

In this example, the tag attribute is set to a parameter value $tag_param:

```
RP/0/RP0/CPU0:router(config-rpl)# set tag $tag_param
```
set traffic-index

To set the traffic index attribute, use the `set traffic-index` command in route-policy configuration mode.

```
set traffic-index {number parameter|ignore}
```

**Syntax Description**
- **number**: Integer value assigned to the traffic index attribute. Range is 1 to 63.
- **parameter**: Parameter name. The parameter name must be preceded with a “$.”
- **ignore**: Specifies that Border Gateway Protocol (BGP) policy accounting is not done.

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `set traffic-index` command to set the traffic index attribute.

**Note**
The `set traffic-index` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

Traffic index is a special attribute for BGP. It is used as an index to a set of counters that are maintained by forwarding hardware. It is also used to track packet and byte counters that are forwarded using routes with specific attributes. These counters can be enabled and disabled on an individual interface basis.

The traffic index attribute can be set only at the table-policy attach point, and can take a value from 1 to 63, or a value of `ignore`. If the traffic index is set to `ignore`, then BGP policy accounting is not done. Parameterization of this value is also supported.

**Task ID**
<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**
In the following example, a policy is created in which the traffic index is set to 10 for all routes that originated in autonomous system 1234:

```
RP/0/RSP0/RP0/CPU0:router(config)# route-policy count-as-1234
RP/0/RSP0/RP0/CPU0:router(config-rpl)# if as-path originates-from '1234' then
RP/0/RSP0/RP0/CPU0:router(config-rpl-if)# set traffic-index 10
RP/0/RSP0/RP0/CPU0:router(config-rpl-if)# else
RP/0/RSP0/RP0/CPU0:router(config-rpl-if)# pass
RP/0/RSP0/RP0/CPU0:router(config-rpl-if)# endif
```
This policy could then be attached using the BGP `table-policy` command. The counters could then be enabled on various interfaces with the appropriate commands.
set vpn-distinguisher

To change the Border Gateway Protocol (BGP) VPN distinguisher attribute, use the **set vpn-distinguisher** command in route-policy configuration mode.

```
set vpn-distinguisher {numberparameter}
```

**Syntax Description**

- `number` Value assigned to a 32-bit unsigned integer. Range is from 1 to 4294967295.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **set vpn-distinguisher** command to change the VPN distinguisher attribute.

**Note**

The **set origin** command can be used as an action statement within an **if** statement. For a list of all action statements available within an **if** statement, see the **if** command.

A VPN distinguisher is used in Layer 3 VPN networks for enhanced individual VPN control and to avoid route target mapping at AS boundaries in inter-AS VPN networks. Route target extended communities are removed at neighbor outbound, and the VPN distinguisher value is applied on the BGP route as an extended community. When the route is received on a neighboring router in another AS, the VPN distinguisher is removed and mapped to a route target extended community.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, the VPN distinguisher attribute is set to 456:

```
RP/0/RP0/CPU0:router(config-rpl)# set vpn-distinguisher 456
```
set weight

To set the weight value for Border Gateway Protocol (BGP) routes, use the **set weight** command in route-policy configuration mode.

```
set weight {numberparameter}
```

**Syntax Description**
- **number**: Number assigned to the weight value for BGP routes. Weight is 16 bits. Range is 0 to 65535.
- **parameter**: Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**
- **Release 6.0**: This command was introduced.

**Usage Guidelines**
Use the **set weight** command to set the weight value for BGP routes.

**Note**
- The **set weight** command can be used as an action statement within an **if** statement. For a list of all action statements available within an **if** statement, see the **if** command.

A weight is a value that can be applied to a route to override the BGP local preference. This is not a BGP attribute announced to BGP peer routers. RPL can be used to set the weight value.

Given two BGP routes with the same network layer reachability information (NLRI), a route with a higher weight is selected, no matter what the values of other BGP attributes may be. However, weight only has significance on the local router. It is not sent from one BGP speaker to another, even within the same autonomous system.

On Cisco routers, if a BGP route is sourced by the local router, its weight is automatically set to 32768; if the BGP route is learned from another router, its weight is automatically set to 0. Thus, by default, locally sourced routes are preferred over BGP learned routes.

**Examples**
In the following example, the weight of the route is set to 10 and then to a parameter value $weight_param:

```
RP/0/RP0/CPU0:router(config-rpl)# set weight 10
RP/0/RP0/CPU0:router(config-rpl)# set weight $weight_param
```
**show rpl**

To display system-wide RPL configuration, use the `show rpl` command in XR EXEC mode.

```
show [running-config] rpl [{maximum {lines configuration-limit|policies policies-limit}|editor {emacs|nano|vim}}]
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>running-config</td>
<td>(Optional) Displays configuration-limit argument.</td>
</tr>
<tr>
<td>maximum</td>
<td>(Optional) Displays the maximum number of lines of configuration and number of policies.</td>
</tr>
<tr>
<td>lines configuration-limit</td>
<td>(Optional) Displays the number of lines to which configuration is limited. Range is 1 to 131072.</td>
</tr>
<tr>
<td></td>
<td>The <code>configuration-limit</code> argument is available if the <code>running-config</code> keyword is specified.</td>
</tr>
<tr>
<td>policies policies-limit</td>
<td>(Optional) Displays the limit on the number of policies. Range is 1 to 5000.</td>
</tr>
<tr>
<td></td>
<td>The <code>configuration-limit</code> argument is available if the <code>running-config</code> keyword is specified.</td>
</tr>
<tr>
<td>editor</td>
<td>(Optional) Specifies the default RPL editor. This keyword is available if the <code>running-config</code> keyword is specified.</td>
</tr>
<tr>
<td>emacs</td>
<td>(Optional) Displays the default RPL editor to Micro Emacs.</td>
</tr>
<tr>
<td>nano</td>
<td>(Optional) Displays the default RPL editor to nano.</td>
</tr>
<tr>
<td>vim</td>
<td>(Optional) Displays the default RPL editor to Vim.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

- **Release 6.0**  This command was introduced.

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following shows the output of the `show running-config rpl` command:
show rpl

RP/0/RP0/CPU0:router# show running-config rpl

extcommunity-set rt ext_comm_set_rt_ex1
   1.2.3.4:34
end-set
!
prefix-set prefix_set_ex1
   10.0.0.0/16 ge 16 le 32,
   0.0.0.0/0 ge 25 le 32,
   0.0.0.0/0
end-set
!
route-policy policy_2
   if destination in prefix_set_ex1 then
      if (community matches-any com_set_ex1) then
         set community (10:666) additive
      endif
      if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
         set community (10:999) additive
      endif
   endif
end-policy
!
show rpl active as-path-set

To display the AS path sets that are referenced by at least one policy that is being used at an attach point, use the `show rpl active as-path-set` command in XR EXEC mode.

`show rpl active as-path-set [detail]`

**Syntax Description**

- **detail** (Optional) Displays the content of the object and all referenced objects for active AS path sets.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl active as-path-set` command to display all AS path sets that are in use in the system and referenced either directly or indirectly at a policy attach point.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast
  !
  neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
      route-policy policy_1 in
  !
  !
  neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
      route-policy policy_2 in
  !
  !
RP/0/RP0/CPU0:router# show rpl route-policy policy_2 detail

prefix-set prefix_set_ex1
  10.0.0.0/16 ge 16 le 32,
  0.0.0.0/0 ge 25 le 32,
  0.0.0.0/0
end-set
!
community-set comm_set_ex1
  65500:1,
```
Given this sample configuration, the `show rpl active as-path-set` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl active as-path-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following as-path-sets are ACTIVE
-------------------------------------
as_path_set_ex1
```

65500:2,
65500:3
end-set

! extcommunity-set rt ext_comm_set_rt_ex1
  1.2.3.4:34
end-set

! route-policy policy_2
  if destination in prefix_set_ex1 then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy

RP/0/RP0/CPU0:router# show rpl route-policy policy_1 detail

prefix-set prefix_set_ex1
  10.0.0.0/16 ge 16 le 32,
  0.0.0.0/0 ge 25 le 32,
  0.0.0.0/0
end-set

! as-path-set as_path_set_ex1
  ios-regex '^_655--$'
  ios-regex '^_65501__$'
end-set

! route-policy policy_1
  if (destination in prefix_set_ex1) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex1) then
    set community (10:333) additive
  endif
end-policy

RP/0/RP0/CPU0:router# show rpl route-policy policy_2 detail

show rpl active community-set

To display the community sets that are referenced by at least one policy that is being used at an attach point, use the show rpl active community-set command in XR EXEC mode.

show rpl active community-set [detail]

**Syntax Description**

- **detail** (Optional) Displays the content of the object and all referenced objects for active community sets.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

Use the show rpl active community-set command to display all community sets that are in use in the system and referenced either directly or indirectly at a policy attach point.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
    address-family ipv4 unicast
    !
    neighbor 10.0.101.2
        remote-as 100
        address-family ipv4 unicast
            route-policy policy_1 in
                !
    neighbor 10.0.101.3
        remote-as 12
        address-family ipv4 unicast
            route-policy policy_2 in
                !
    !

RP/0/RP0/CPU0:router# show rpl route-policy policy_2 detail

prefix-set prefix_set_ex1
    10.0.0.0/16 ge 16 le 32,
    0.0.0.0/0 ge 25 le 32,
    0.0.0.0/0
end-set
!
community-set comm_set_ex1
```
Given this sample configuration, the `show rpl active community-set` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl active community-set
ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following community-sets are ACTIVE
---------------------------------------
comm_set_ex1
```

```text
route-policy policy_2
  if destination in prefix_set_ex1 then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy
```
show rpl active extcommunity-set

To display the extended community sets for cost, route target (RT), and Site-of-Origin (SoO) that are referenced by at least one route policy used at an attach point, use the `show rpl active extcommunity-set` command in XR EXEC mode.

```
show rpl active extcommunity-set [{cost|rt|soo}] [detail]
```

**Syntax Description**

- **cost** (Optional) Displays all extended community cost sets.
- **rt** (Optional) Displays all extended community RT sets.
- **soo** (Optional) Displays all extended community SoO sets.
- **detail** (Optional) Displays the content of the object and all referenced objects for active extended community sets.

**Command Default**

All extended community sets are displayed.

**Command Modes**

XR EXEC mode

**Command History**

- **Release** Release 6.0  This command was introduced.

**Usage Guidelines**

Use the `show rpl active extcommunity-set` command to display all extended community sets that are in use in the system and referenced either directly or indirectly at a policy attach point.

**Examples**

This example shows the following sample configuration:

```
router bgp 2
 address-family ipv4 unicast
!
neighbor 10.0.101.2
 remote-as 100
 address-family ipv4 unicast
 route-policy policy_1 in
!
!
neighbor 10.0.101.3
 remote-as 12
 address-family ipv4 unicast
 route-policy policy_2 in
!
!
```
Given this sample configuration, the `show rpl active extcommunity-set` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl active extcommunity-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
```
UNUSED -- Not attached (directly or indirectly) and not referenced

The following extcommunity-sets are ACTIVE
------------------------------------------
ext_comm_set_rt_ex1
show rpl active prefix-set

To display the prefix sets that are referenced by at least one policy that is being used at an attach point, use the `show rpl active prefix-set` command in XR EXEC mode.

`show rpl active prefix-set [detail]`

**Syntax Description**

- **detail** (Optional) Displays the content of the object and all referenced objects for active prefix sets.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl active prefix-set` command to display all prefix sets that are in use in the system and referenced either directly or indirectly at a policy attach point.

**Examples**

This example shows the following sample configuration:

```bash
router bgp 2
  address-family ipv4 unicast
    !
  neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
      route-policy policy_1 in
        !
    neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
      route-policy policy_2 in
        !
    !

RP/0/RP0/CPU0:router# show rpl route-policy policy_2 detail

prefix-set prefix_set_ex1
  10.0.0.0/16 ge 16 le 32,
  0.0.0.0/0 ge 25 le 32,
  0.0.0.0/0
end-set
```

Routing Command Reference for Cisco NCS 5000 Series Routers
The following example displays active prefix sets:

RP/0/RP0/CPU0:router# show rpl active prefix-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following prefix-sets are ACTIVE
------------------------------------
prefix_set_1

prefix-set prefix_set_1
  10.0.0.0/16 ge 16 le 32,
  0.0.0.0/0 ge 25 le 32,
  0.0.0.0/0
end-set
!
route-policy policy_1
  if (destination in prefix_set_1) then
    set local-preference 100
  endif
  if (as-path in as_path_set_1) then
    set community (10:333) additive
  endif
end-policy
!

The following example displays active prefix sets:

RP/0/RP0/CPU0:router# show rpl active prefix-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following prefix-sets are ACTIVE
------------------------------------
prefix_set_1
show rpl active rd-set

To display the route distinguisher (RD) sets that are referenced by at least one policy that is being used at an attach point, use the **show rpl active rd-set** command in XR EXEC mode.

```
show rpl active rd-set [detail]
```

### Syntax Description

**detail** (Optional) Displays the content of the object and all referenced objects for active route policies.

### Command Default

No default behavior or values

### Command Modes

XR EXEC mode

### Command History

**Release** Modification

Release 6.0 This command was introduced.

### Usage Guidelines

Use the **show rpl active rd-set** command to display all RD sets that are in use in the system and that are referenced either directly or indirectly at a policy attach point.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

### Examples

This example shows the following sample configuration:

```
rd-set rdset1
  10:151,
  100.100.100.1:153,
  100.100.100.62/31:63
end-set
!
rd-set rdset2
  10:152,
  100.100.100.1:154,
  100.100.100.62/31:89
end-set
!
route-policy rdsetmatch
  if rd in rdset1 then
    set community (10:112)
  elseif rd in rdset2 then
    set community (10:223)
  endif
end-policy
!
router bgp 10
  bgp router-id 10.0.0.1
  address-family vpnv4 unicast
  neighbor 10.10.10.1
    remote-as 10
    address-family ipv4 unicast
```
route-policy rdsetmatch in
!
!

Given this sample configuration, the **show rpl active rd-set** command displays the following information:

RP/0/RP0/CPU0:router# **show rpl active rd-set**

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following rd-sets are ACTIVE
------------------------------------------
rdset1
rdset2
show rpl active route-policy

To display the route policies that are referenced by at least one policy that is being used at an attach point, use the show rpl active route-policy command in XR EXEC mode.

show rpl active route-policy [detail]

Syntax Description

- detail (Optional) Displays the content of the object and all referenced objects for active route policies.

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

Use the show rpl active route-policy command to display all policies that are in use in the system and that are referenced either directly or indirectly at a policy attach point.

Task ID

- Task ID: route-policy
- Operations: read

Examples

This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast

neighbor 10.0.101.2
  remote-as 100
  address-family ipv4 unicast
  route-policy policy_1 in

neighbor 10.0.101.3
  remote-as 12
  address-family ipv4 unicast
  route-policy policy_2 in

RP/0/RP0/CPU0:router# show rpl route-policy policy_1

route-policy policy_1
  if (destination in prefix_set_ex1) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex1) then
    set community (10:333) additive
  endif
```
end-policy
!
RP/0/RP0/CPU0:router# show rpl route-policy policy_2

route-policy policy_2
  if destination in prefix_set_ex1 then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy
!

Given this sample configuration, the **show rpl active route-policy** command displays the following information:

RP/0/RP0/CPU0:router# show rpl active route-policy

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following policies are (ACTIVE)
-----------------------------------
policy_1
policy_2
show rpl as-path-set

To display the contents of AS path sets, use the show rpl as-path-set command in XR EXEC mode.

`show rpl as-path-set [{name|states|brief}]`

**Syntax Description**
- `name` (Optional) Name of the AS path set.
- `states` (Optional) Displays all unused, inactive, and active states.
- `brief` (Optional) Limits the display to a list of the names of all AS path sets without their configurations.

**Command Default**
No default behavior or values

**Command Modes**
XR EXEC mode

**Command History**
<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the optional `brief` keyword to limit the display to a list of the names of all AS path sets without their configurations.

**Examples**
This example shows the following sample configuration:

```
RP/0/RP0/CPU0:router# show rpl route-policy policy_1

route-policy policy_1
  if (destination in prefix_set_ex1) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex1) then
    set community (10:333) additive
  endif
end-policy
```

Given this sample configuration, the `show rpl as-path-set as_path_set_ex1` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl as-path-set as_path_set_ex1

as-path-set as_path_set_ex1
  ios-regex '^_65500_$',
  ios-regex '^_65501_$'
end-set
```
show rpl as-path-set attachpoints

To display all of the policies used at an attach point that reference the named AS path set, use the `show rpl as-path-set attachpoints` command in XR EXEC mode.

```
show rpl as-path-set name attachpoints
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of an AS path set.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl as-path-set attachpoints` command to display all policies used at an attach point that reference the named set either directly or indirectly.

The AS path set name is required.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
    address-family ipv4 unicast
    !
neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
    route-policy policy_1 in
    !
neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
    route-policy policy_2 in
    !

RP/0/RP0/CPU0:router# show rpl route-policy policy_1
route-policy policy_1
    if (destination in prefix_set_ex1) then
    set local-preference 100
    endif
    if (as-path in as_path_set_ex1) then
```
set community (10:333) additive
endif
end-policy
!
RP/0/RP0/CPU0:router# show rpl route-policy policy_2

route-policy policy_2
  if (destination in prefix_set_ex1) then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy
!

Given this sample configuration, the `show rpl as-path-set as_path_set_ex1 attachpoints` command displays the following information:

RP/0/RP0/CPU0:router# show rpl as-path-set as_path_set_ex1 attachpoints

BGP Attachpoint:Neighbor

<table>
<thead>
<tr>
<th>Neighbor/Group</th>
<th>type</th>
<th>afi/safi</th>
<th>in/out</th>
<th>referring policy</th>
<th>attached policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.101.2</td>
<td>--</td>
<td>IPv4/uni</td>
<td>in</td>
<td>policy_1</td>
<td>policy_1</td>
</tr>
<tr>
<td>10.0.101.3</td>
<td>--</td>
<td>IPv4/uni</td>
<td>in</td>
<td>policy_2</td>
<td>policy_2</td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

Table 110: `show rpl as-path-set attachpoints` Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BGP Attachpoint</td>
<td>Location of the attach point.</td>
</tr>
<tr>
<td>Neighbor/Group</td>
<td>IP address of the attach point</td>
</tr>
<tr>
<td></td>
<td>on the neighbor.</td>
</tr>
<tr>
<td>type</td>
<td>Displays the address family mode.</td>
</tr>
<tr>
<td>afi/safi</td>
<td>Address family identifier or</td>
</tr>
<tr>
<td></td>
<td>subsequent address family</td>
</tr>
<tr>
<td></td>
<td>identifier.</td>
</tr>
<tr>
<td>in/out</td>
<td>Import or export policy.</td>
</tr>
<tr>
<td>referring policy</td>
<td>Policy that refers to the AS</td>
</tr>
<tr>
<td></td>
<td>path set.</td>
</tr>
<tr>
<td>attached policy</td>
<td>Policy used at the attach point.</td>
</tr>
</tbody>
</table>
show rpl as-path-set references

To list all of the policies that reference the named AS path set, use the `show rpl as-path-set references` command in XR EXEC mode.

`show rpl as-path-set name references [brief]`

**Syntax Description**

- `name` Name of the prefix set.
- `brief` (Optional) Limits the output to just the brief table and not the detailed information for the named AS path set.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

- **Release 6.0** This command was introduced.

**Usage Guidelines**

Use the `show rpl as-path-set references` command to display all policies that reference the named AS path set either directly or indirectly.

Use the optional `brief` keyword to limit the output to just a summary table and not the detailed information for the AS path set.

**Examples**

This example shows the following sample configuration:

```
router bgp 2
    address-family ipv4 unicast
    !
    neighbor 10.0.101.2
        remote-as 100
        address-family ipv4 unicast
            route-policy policy_1 in
                !
            !
            RP/0/RP0/CPU0:router# show rpl route-policy policy_1

route-policy policy_1
    if (destination in prefix_set_ex1) then
        set local-preference 100
    endif
    if (as-path in as_path_set_ex1) then
        set community (10:333) additive
    endif
```
end-policy

Given this sample configuration, the `show rpl as-path-set as_path_set_ex1 references` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl as-path-set as_path_set_ex1 references
```

**Usage**
- **Direct** -- Reference occurs in this policy
- **Indirect** -- Reference occurs via an apply statement

**Status**
- **UNUSED** -- Policy is not in use at an attachpoint (unattached)
- **ACTIVE** -- Policy is actively used at an attachpoint
- **INACTIVE** -- Policy is applied by an unattached policy

<table>
<thead>
<tr>
<th>Usage/Status</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>1</td>
</tr>
<tr>
<td>Indirect</td>
<td>0</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>1</td>
</tr>
<tr>
<td>INACTIVE</td>
<td>0</td>
</tr>
<tr>
<td>UNUSED</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>route-policy</th>
<th>usage</th>
<th>policy status</th>
</tr>
</thead>
<tbody>
<tr>
<td>policy_1</td>
<td>Direct</td>
<td>ACTIVE</td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

**Table 111: show rpl as-path-set references Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usage/Status</td>
<td>Displays the usage and status of all policies that reference the AS path set. Values for usage are Direct or Indirect. Values for policy status are ACTIVE, INACTIVE, or UNUSED.</td>
</tr>
<tr>
<td>count</td>
<td>Number of policies that match each usage and status option.</td>
</tr>
<tr>
<td>route-policy</td>
<td>Name of the route policies that reference the AS path set.</td>
</tr>
<tr>
<td>usage</td>
<td>Type of usage for the policy.</td>
</tr>
<tr>
<td>policy status</td>
<td>Status of the policy.</td>
</tr>
</tbody>
</table>
show rpl community-set

To display the configuration of community sets, use the **show rpl community-set** command in XR EXEC mode.

**show rpl community-set** [{*name*|*states*|*brief*}]

**Syntax Description**

<table>
<thead>
<tr>
<th></th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>name</em></td>
<td>(Optional) Name of the community set.</td>
</tr>
<tr>
<td><em>states</em></td>
<td>(Optional) Shows all unused, inactive, and active states.</td>
</tr>
<tr>
<td><em>brief</em></td>
<td>(Optional) Limits the display to a list of the names of all community sets without their configurations.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the optional **brief** keyword to limit the display to a list of the names of community sets without their configurations.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

The following is the sample output of the show rpl community-set command with graceful maintenance feature attributes displayed:

```
RP/0/0/CPU0:R5#show rpl community-set
Thu Jan 29 17:55:04.792 PST
Listing for all Community Set objects

community-set gshut
  graceful-shutdown
end-set
```

**Examples**

This example shows the following sample configuration:

```
route-policy policy_4
  if (destination in prefix_set_ex2) then
    if (community matches-any comm_set_ex2) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex2) then
      set community (10:999) additive
    endif
  endif
```
Given this sample configuration, the `show rpl community-set comm_set_ex2` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl community-set comm_set_ex2

community-set comm_set_ex2
  65501:1,
  65501:2,
  65501:3
end-set
```
show rpl community-set attachpoints

To display all the policies used at an attach point that reference the named community set, use the `show rpl community-set attachpoints` command in XR EXEC mode.

```
show rpl community-set name attachpoints
```

**Syntax Description**

- `name` Name of a community set.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl community-set attachpoints` command to display all the policies used at an attach point that reference the named community set either directly or indirectly.

The community set name is required.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>route-policy read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast
  neighbor 10.0.101.3
  remote-as 12
  address-family ipv4 unicast
    route-policy policy_2 in
      !
      !
  route-policy policy_2
    if destination in prefix_set_ex1 then
      if (community matches-any comm_set_ex1) then
        set community (10:666) additive
      endif
      if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
        set community (10:999) additive
      endif
    endif
end-policy
```
Given this sample configuration, the `show rpl community-set attachpoints` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl community-set ext_comm_set_rt_ex1 attachpoints
```

```
BGP Attachpoint:Neighbor
Neighbor/Group  type  afi/safi  in/out  referring policy  attached policy
----------------------------------------------------------------------------
10.0.101.3      --  IPv4/uni  in    policy_2          policy_2
```

This table describes the significant fields shown in the display.

### Table 112: show rpl community-set attachpoints Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BGP Attachpoint</td>
<td>Location of the attach point.</td>
</tr>
<tr>
<td>Neighbor/Group</td>
<td>IP address of the attach point on the neighbor.</td>
</tr>
<tr>
<td>type</td>
<td>Displays the address family mode.</td>
</tr>
<tr>
<td>afi/safi</td>
<td>Address family identifier or subsequent address family identifier.</td>
</tr>
<tr>
<td>in/out</td>
<td>Import or export policy.</td>
</tr>
<tr>
<td>referring policy</td>
<td>Policy that refers to the AS path set.</td>
</tr>
<tr>
<td>attached policy</td>
<td>Policy used at the attach point.</td>
</tr>
</tbody>
</table>
show rpl community-set references

To list all the policies that reference the named community set, use the show rpl community-set references command in XR EXEC mode.

```
show rpl community-set name references [brief]
```

**Syntax Description**
- `name` Name of a community set.
- `brief` (Optional) Limits the output to just the summary table and not the detailed information for the community set.

**Command Default**
No default behavior or values

**Command Modes**
XR EXEC mode

**Command History**
- **Release** Modification
  - Release 6.0 This command was introduced.

**Usage Guidelines**
Use the show rpl community-set references command to display all the policies that reference the named community set.

Use the optional `brief` keyword to limit the output to just a summary table and not the detailed information for the community set.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**
This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast
    !
  neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
      route-policy policy_2 in
        !
        !
route-policy policy_2
  if (destination in prefix_set_ex1) then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set Rt_ex1) then
      set community (10:999) additive
    endif
  endif
```
Given this sample configuration, the **show rpl extcommunity-set comm_set_ex1 references** command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl extcommunity-set comm_set_ex1 references
```

<table>
<thead>
<tr>
<th>Usage/Status</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>1</td>
</tr>
<tr>
<td>Indirect</td>
<td>0</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>1</td>
</tr>
<tr>
<td>INACTIVE</td>
<td>0</td>
</tr>
<tr>
<td>UNUSED</td>
<td>0</td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

**Table 113: show rpl community-set references Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usage/Status</td>
<td>Displays the usage and status of all policies that reference the community set.</td>
</tr>
<tr>
<td></td>
<td>Values for usage are Direct or Indirect.</td>
</tr>
<tr>
<td></td>
<td>Values for status are ACTIVE, INACTIVE, and UNUSED.</td>
</tr>
<tr>
<td>count</td>
<td>Number of policies that match each usage and status option.</td>
</tr>
<tr>
<td>route-policy</td>
<td>Name of the route policies that reference the community set.</td>
</tr>
<tr>
<td>usage</td>
<td>Type of usage for the policy.</td>
</tr>
<tr>
<td>policy status</td>
<td>Status of the policy.</td>
</tr>
</tbody>
</table>
# show rpl extcommunity-set

To display the configuration of extended community sets, use the `show rpl extcommunity-set` command in XR EXEC mode.

```
show rpl extcommunity-set [name [{attachpoints|references}] [{cost|rt|soo}] [name] [brief] [states]
```

## Syntax Description

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>(Optional) Name of the community set.</td>
</tr>
<tr>
<td>attachpoints</td>
<td>(Optional) Displays all attach points for this community set.</td>
</tr>
<tr>
<td>references</td>
<td>(Optional) Displays all policies that use this community set.</td>
</tr>
<tr>
<td>cost</td>
<td>(Optional) Displays all extended community cost sets.</td>
</tr>
<tr>
<td>rt</td>
<td>(Optional) Displays all extended community RT sets.</td>
</tr>
<tr>
<td>soo</td>
<td>(Optional) Displays all extended community SoO sets.</td>
</tr>
<tr>
<td>brief</td>
<td>(Optional) Limits the display to a list of the names of all extended community sets without their configurations.</td>
</tr>
<tr>
<td>states</td>
<td>(Optional) Displays all unused, inactive, and active states.</td>
</tr>
</tbody>
</table>

## Command Default

- If an attachpoint or reference is not specified, all configured extended community sets are displayed.
- If a cost, RT, or SoO sets is not specified, all configured extended community sets are displayed.

## Command Modes

- XR EXEC mode

## Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

## Usage Guidelines

Use the optional `brief` keyword to limit the display to a list of the names of extended community sets without their configurations.

## Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

## Examples

In the following example, the configuration of an extended community is displayed for the RT community set named `ext_comm_set_rt_ex1`:

```
RP/0/RP0/CPU0:router# show rpl extcommunity-set rt ext_comm_set_rt_ex1
ext_comm_set_rt_ex1
  1.2.3.4:34
end-set
```
In the following example, the configuration of an extended community is displayed with all RT set objects:

RP/0/RP0/CPU0:router# show rpl extcommunity-set rt

Listing for all Extended Community RT Set objects

extcommunity-set rt extrt1
  66:60001
  end-set
!

extcommunity-set rt rtset1
  10:615,
  10:6150,
  15.15.15.15:15
  end-set
!

extcommunity-set rt rtset3
  11:11,
  11.1.1.1:3
  end-set
!

extcommunity-set rt extsoo1
  66:70001
  end-set
!

extcommunity-set rt rtsetl1
  100:121,
  100:122,
  100:123,
  100:124,
  100:125,
  100:126,
  100:127,
  100:128,
  7.7.7.7:21
  end-set
!

In the following example, the configuration of an extended community is displayed with all cost set objects:

RP/0/RP0/CPU0:router# show rpl extcommunity-set cost

Listing for all Extended Community COST Set objects

extcommunity-set cost costset1
  IGP:90:914,
  Pre-Bestpath:91:915
  end-set
!

extcommunity-set cost costset2
  IGP:92:916,
  Pre-Bestpath:93:917,
  IGP:94:918,
  Pre-Bestpath:95:919
  end-set
!
In the following example, the configuration of an extended community is displayed with all SoO set objects:

Extended Community SOO Set objects

extcommunity-set soo sooset1
  10:151,
  100.100.100.1:153
end-set
!

extcommunity-set soo sooset3
  11:11,
  11.1.1.1:3
end-set
!
show rpl inactive as-path-set

To display the AS path sets that are referenced by a policy but not in any policy that is used at an attach point, use the `show rpl inactive as-path-set` command in XR EXEC mode.

```
show rpl inactive as-path-set [detail]
```

### Syntax Description

- **detail** (Optional) Displays the content of the object and all referenced objects for inactive AS path sets.

### Command Default

No default behavior or values

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Use the `show rpl inactive as-path-set` command to display all AS path sets that are not in use at an attach point either directly or indirectly but are referenced by at least one policy in the system.

### Task ID

- **read**

### Examples

This example shows the following sample configuration:

```
routerr bgp 2
    address-family ipv4 unicast
    !
    neighbor 10.0.101.2
        remote-as 100
        address-family ipv4 unicast
        route-policy policy_1 in
        !
    neighbor 10.0.101.3
        remote-as 12
        address-family ipv4 unicast
        route-policy policy_2 in
        !
    !
    route-policy sample
        if (destination in sample) then
            drop
        endif
    end-policy
    !
    route-policy policy_1
        if (destination in prefix_set_ex1) then
            set local-preference 100
        endif
```
if (as-path in as_path_set_ex1) then
  set community (10:333) additive
endif
end-policy
!
route-policy policy_2
  if destination in prefix_set_ex1 then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy
!
route-policy policy_3
  if (destination in prefix_set_ex2) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex2) then
    set community (10:333) additive
  endif
end-policy
!
route-policy policy_4
  if (destination in prefix_set_ex2) then
    if (community matches-any comm_set_ex2) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex2) then
      set community (10:999) additive
    endif
  endif
end-policy
!
route-policy policy_5
  apply sample1
  apply policy_3
end-policy

Given this sample configuration, the **show rpl inactive as-path-set** command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl inactive as-path-set
ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following as-path-sets are INACTIVE
---------------------------------------
as_path_set_ex2
```
show rpl inactive community-set

To display the community sets that are referenced by a policy but not any policy that is used at an attach point, use the `show rpl inactive community-set` command in XR EXEC mode.

```
show rpl inactive community-set [detail]
```

**Syntax Description**

- `detail` (Optional) Displays the content of the object and all referenced objects for inactive community sets.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
</table>
| 6.0     | This command was introduced.

**Usage Guidelines**

Use the `show rpl inactive community-set` command to display all community sets that are not in use at an attach point either directly or indirectly but are referenced by at least one policy in the system.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
routing bgp 2
address-family ipv4 unicast

neighbor 10.0.101.2
remote-as 100
address-family ipv4 unicast
route-policy policy_1 in

neighbor 10.0.101.3
remote-as 12
address-family ipv4 unicast
route-policy policy_2 in

route-policy sample2
  if (destination in sample2) then
    drop
  endif
end-policy

route-policy policy_1
  if (destination in prefix_set_ex1) then
    set local-preference 100
  endif
```
if (as-path in as_path_set_ex1) then
    set community (10:333) additive
endif
end-policy
!
route-policy policy_2
    if destination in prefix_set_ex1 then
        if (community matches-any comm_set_ex1) then
            set community (10:666) additive
        endif
        if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
            set community (10:999) additive
        endif
    endif
end-policy
!
route-policy policy_3
    if (destination in prefix_set_ex2) then
        set local-preference 100
    endif
    if (as-path in as_path_set_ex2) then
        set community (10:333) additive
    endif
end-policy
!
route-policy policy_4
    if (destination in prefix_set_ex2) then
        if (community matches-any comm_set_ex2) then
            set community (10:666) additive
        endif
        if (extcommunity matches-any ext_comm_set_rt_ex2) then
            set community (10:999) additive
        endif
    endif
end-policy
!
route-policy policy_5
    apply sample2
    apply policy_3
end-policy

Given this sample configuration, the **show rpl inactive community-set** command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl inactive community-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following community-sets are INACTIVE

------------------------------------------
comm_set_ex2
```
show rpl inactive extcommunity-set

To display the extended community sets that are referenced by a policy but not in any policy that is used at an attach point, use the `show rpl inactive extcommunity-set` command in XR EXEC mode.

```
show rpl inactive extcommunity-set [detail]
```

**Syntax Description**
- `detail` (Optional) Displays the content of the object and all referenced objects for inactive extended community sets.

**Command Default**
No default behavior or values

**Command Modes**
XR EXEC mode

**Command History**
- **Release** 6.0  This command was introduced.

**Usage Guidelines**
Use the `show rpl inactive extcommunity-set` command to display all extended community sets that are not in use at an attach point either directly or indirectly but are referenced by at least one policy in the system.

**Examples**
This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast
  !
  neighbor 10.0.101.2
  remote-as 100
  address-family ipv4 unicast
    route-policy policy_1 in
    !
  !
  neighbor 10.0.101.3
  remote-as 12
  address-family ipv4 unicast
    route-policy policy_2 in
    !
  !
  route-policy sample3
    if (destination in sample3) then
      drop
    endif
  end-policy
  !
  route-policy policy_1
    if (destination in prefix_set_ex1) then
      set local-preference 100
```

Routing Command Reference for Cisco NCS 5000 Series Routers
Given this sample configuration, the `show rpl inactive extcommunity-set` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl inactive extcommunity-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following extcommunity-sets are INACTIVE
--------------------------------------------
ext_comm_set_rt_ex2
```

show rpl inactive prefix-set

To display the prefix sets that are referenced by a policy but not in any policy that is used at an attach point, use the `show rpl inactive prefix-set` command in XR EXEC mode.

```
show rpl inactive prefix-set [detail]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>detail (Optional) Displays the content of the object and all referenced objects for inactive prefix sets.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Default</td>
<td>No default behavior or values</td>
</tr>
<tr>
<td>Command Modes</td>
<td>XR EXEC mode</td>
</tr>
<tr>
<td>Command History</td>
<td>Release 6.0  This command was introduced.</td>
</tr>
<tr>
<td>Usage Guidelines</td>
<td>Use the <code>show rpl inactive prefix-set</code> command to display all prefix sets that are not in use at an attach point either directly or indirectly but are referenced by at least one policy in the system.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast
    !
  neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
      route-policy policy_1 in
      !
    neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
      route-policy policy_2 in
      !
    !
  route-policy sample4
    if (destination in sample4) then
      drop
    endif
  end-policy
  !
  route-policy policy_1
    if (destination in prefix_set_ex1) then
      set local-preference 100
    endif
```
if (as-path in as_path_set_ex1) then
    set community (10:333) additive
endif
end-policy
!
route-policy policy_2
    if destination in prefix_set_ex1 then
        if (community matches-any comm_set_ex1) then
            set community (10:666) additive
        endif
        if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
            set community (10:999) additive
        endif
    endif
end-policy
!
route-policy policy_3
    if (destination in prefix_set_ex2) then
        set local-preference 100
    endif
    if (as-path in as_path_set_ex2) then
        set community (10:333) additive
    endif
end-policy
!
route-policy policy_4
    if (destination in prefix_set_ex2) then
        if (community matches-any comm_set_ex2) then
            set community (10:666) additive
        endif
        if (extcommunity matches-any ext_comm_set rt_ex2) then
            set community (10:999) additive
        endif
    endif
end-policy
!
route-policy policy_5
    apply sample4
    apply policy_3
end-policy

Given this sample configuration, the `show rpl inactive prefix-set` command displays the following information:

RP/0/RP0/CPU0:router# show rpl inactive prefix-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following prefix-sets are INACTIVE
--------------------------------------
sample4
prefix_set_ex2
**show rpl inactive rd-set**

To display the route distinguisher (RD) sets that are referenced by a policy but not in any policy that is used at an attach point, use the `show rpl inactive rd-set` command in XR EXEC mode.

```
show rpl inactive rd-set [detail]
```

**Syntax Description**

- `detail` (Optional) Displays the content of the object and all referenced objects for inactive RD sets.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl inactive rd-set` command to display all RD sets that are not in use at an attach point either directly or indirectly but are referenced by at least one policy in the system.

**Examples**

This example shows the following sample configuration:

```plaintext
rd-set rdset1
  10:151,
  100.100.100.1:153,
  100.100.100.62/31:63
end-set
!
rd-set rdset2
  10:152,
  100.100.100.1:154,
  100.100.100.62/31:89
end-set
!
```

Given this sample configuration, the `show rpl inactive rd-set` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl inactive rd-set

ACTIVE -- Referenced by at least one policy which is attached INACTIVE -- Only referenced by policies which are not attached UNUSED -- Not attached (directly or indirectly) and not referenced

The following rd-sets are INACTIVE
------------------------------------------
rdset1
```

Routing Command Reference for Cisco NCS 5000 Series Routers

728
show rpl inactive rd-set

rdset2
show rpl inactive route-policy

To display the route policies that are referenced by a policy but not in any policy that is used at an attach point, use the **show rpl inactive route-policy** command in XR EXEC mode.

```
show rpl inactive route-policy [detail]
```

**Syntax Description**

```
detail  (Optional) Displays the content of the object and all referenced objects for inactive route policies.
```

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

```
Release      Modification
------------- -----------------------
Release 6.0    This command was introduced.
```

**Usage Guidelines**

Use the **show rpl inactive route-policy** command to display all policies that are not in use at an attach point either directly or indirectly but are referenced by at least one other policy in the system.

**Task ID**

```
Task ID  Operations
---------  -------------------
route-policy  read
```

**Examples**

This example shows the following sample configuration:

```
routing bgp 2
  address-family ipv4 unicast
  !
  neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
      route-policy policy_1 in
    !
  neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
      route-policy policy_2 in
    !
  !
  route-policy sample3
    if (destination in sample3) then
      drop
    endif
  end-policy
  !
  route-policy policy_1
    if (destination in prefix_set_ex1) then
      set local-preference 100
    end
```
if (as-path in as_path_set_ex1) then
    set community (10:333) additive
endif
end-policy
!
route-policy policy_2
    if destination in prefix_set_ex1 then
        if (community matches-any comm_set_ex1) then
            set community (10:666) additive
        endif
        if (extcommunity rt matches-any ext_comm_set_rt_ex1) then
            set community (10:999) additive
        endif
    endif
end-policy
!
route-policy policy_3
    if (destination in prefix_set_ex2) then
        set local-preference 100
    endif
    if (as-path in as_path_set_ex2) then
        set community (10:333) additive
    endif
end-policy
!
route-policy policy_4
    if (destination in prefix_set_ex2) then
        if (community matches-any comm_set_ex2) then
            set community (10:666) additive
        endif
        if (extcommunity matches-any ext_comm_set_rt_ex2) then
            set community (10:999) additive
        endif
    endif
end-policy
!
route-policy policy_5
    apply sample3
    apply policy_3
end-policy

Given this sample configuration, the show rpl inactive route-policy command displays the following information:

RP/0/RP0/CPU0:router# show rpl inactive route-policy

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following policies are (INACTIVE)
-------------------------------------
sample3
policy_3
show rpl maximum

To display the maximum limits for lines of configuration and number of policies, use the `show rpl maximum` command in XR EXEC mode.

```
show rpl maximum [{lines|policies}]
```

**Syntax Description**

- `lines` (Optional) Displays the number of lines of configuration limit.
- `policies` (Optional) Displays the number of policies limit.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl maximum` command to display the current total, current limit, and maximum limit for lines of configuration and policies.

Use the optional `lines` keyword to limit the display to the number of lines of configuration limits. Use the optional `policies` keyword to limit the display to the number of policies limits.

**Examples**

The following example shows sample output from the `show rpl maximum` command:

```
RP/0/RP0/CPU0:router# show rpl maximum
                          Current  Current  Max
                          Total    Limit    Limit
-----------------------------------------------
Lines of configuration  3        65536    131072
Policies                1         3500     5000
Compiled policies size (kB) 0
```

**Table 114: show rpl maximum Field Descriptions, on page 732** describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lines of configuration</td>
<td>Displays the current total, current limit, and maximum limit of lines for the policy.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------------------------------------------------------</td>
</tr>
<tr>
<td>Policies</td>
<td>Displays the current total, current limit, and maximum limit of policies.</td>
</tr>
<tr>
<td>Compiled policies size (kB)</td>
<td>Displays the current compiled total for policies in kilobytes.</td>
</tr>
</tbody>
</table>
show rpl policy-global references

To display policy-global definitions, use the `show rpl policy-global references` command in XR EXEC mode.

```
show rpl policy-global references [brief]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>brief</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(Optional) Limits the display to a list of the policy names.</td>
</tr>
</tbody>
</table>

### Command Default

No default behavior or values

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

No specific guidelines impact the use of this command.

### Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>route-policy read</td>
</tr>
</tbody>
</table>

### Examples

This example shows the following sample configuration:

```
policy-global
  infinity '16'
end-global
!
route-policy set-rip-unreachable
  set rip-metric $infinity
end-policy
!
```

Given this sample configuration, the `show rpl policy-global references` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl policy-global references
Usage Direct -- Reference occurs in this policy Usage Indirect -- Reference occurs via an apply statement
Status UNUSED -- Policy is not in use at an attachpoint (unattached) Status ACTIVE -- Policy is actively used at an attachpoint Status INACTIVE -- Policy is applied by an unattached policy

Usage/Status count
----------------------------------
Direct                              1
Indirect                            0
```
<table>
<thead>
<tr>
<th>Usage</th>
<th>Status</th>
<th>Route-policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>UNUSED</td>
<td>set-rip-unreachable</td>
</tr>
</tbody>
</table>

ACTIVE: 0
INACTIVE: 0
UNUSED: 1
show rpl prefix-set

To display the configuration of prefix sets, use the `show rpl prefix-set` command in XR EXEC mode.

```
show rpl prefix-set [{name|states|brief}]
```

### Syntax Description

- **name** (Optional) Name of the prefix set.
- **states** (Optional) Shows all unused, inactive, and active states.
- **brief** (Optional) Limits the display to a list of the names of all extended community sets without their configurations.

### Command Default

No default behavior or values

### Command Modes

XR EXEC mode

### Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

### Usage Guidelines

Because sets cannot hierarchically reference other sets or policies, no `detail` keyword exists as with the `show rpl policy` command.

### Examples

In the following example, the configuration of prefix set pset1 is displayed:

```
RP/0/RP0/CPU0:router# show rpl prefix-set pset1
prefix-set pset1
10.0.0.1/0,
10.0.0.2/0 ge 25 le 32,
10.0.0.5/8 ge 8 le 32,
10.168.0.0/16 ge 16 le 32,
172.16.0.9/20 ge 20 le 32,
192.168.0.5/20 ge 20 le 32
end-set
```
show rpl prefix-set attachpoints

To display all the policies used at an attach point that reference the named prefix set, use the `show rpl prefix-set attachpoints` command in XR EXEC mode.

```
show rpl prefix-set name attachpoints
```

**Syntax Description**

`name` Name of a prefix set.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl prefix-set attachpoints` command to display all the policies used at an attach point that reference the named prefix set either directly or indirectly.

The prefix set name is required.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast
  !
neighbor 10.0.101.2
  remote-as 100
  address-family ipv4 unicast
  route-policy policy_1 in
  !
!
neighbor 10.0.101.3
  remote-as 12
  address-family ipv4 unicast
  route-policy policy_2 in
  !
!
route-policy policy_1
  if (destination in prefix_set_ex1) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex1) then
    set community (10:333) additive
  endif
end-policy
```

Routing Command Reference for Cisco NCS 5000 Series Routers

737
route-policy policy_2
  if (destination in prefix_set_ex1) then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy

Given this sample configuration, the `show rpl prefix-set prefix_set_ex1 attachpoints` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl prefix-set prefix_set_ex1 attachpoints
BGP Attachpoint:Neighbor
Neighbor/Group type  afi/safi  in/out  referring policy attached policy
----------------------------------------------------------------------------
10.0.101.2  -- IPv4/uni  in  policy_1  policy_1
10.0.101.3  -- IPv4/uni  in  policy_2  policy_2
```

This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BGP Attachpoint</td>
<td>Location of the attach point.</td>
</tr>
<tr>
<td>Neighbor/Group</td>
<td>IP address of the attach point on the neighbor.</td>
</tr>
<tr>
<td>type</td>
<td>Address family mode.</td>
</tr>
<tr>
<td>afi/safi</td>
<td>Address family identifier or subsequent address family identifier.</td>
</tr>
<tr>
<td>in/out</td>
<td>Import or export policy.</td>
</tr>
<tr>
<td>referring policy</td>
<td>Policy that refers to the AS path set.</td>
</tr>
<tr>
<td>attached policy</td>
<td>Policy used at the attach point.</td>
</tr>
</tbody>
</table>
show rpl prefix-set references

To list all the policies that reference the named prefix set, use the show rpl prefix-set references command in XR EXEC mode.

show rpl prefix-set name references [brief]

Syntax Description

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of the prefix set.</td>
</tr>
<tr>
<td>brief</td>
<td>(Optional) Limits the output to just a summary table and not the detailed information for the named prefix set.</td>
</tr>
</tbody>
</table>

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

Usage Guidelines

Use the show rpl prefix-set references command to list all the policies that reference the named prefix set. Use the optional brief keyword to limit the output to just a summary table and not the detailed information for the named prefix set.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

Examples

This example shows the following sample configuration:

```plaintext
prefix-set ten-net
  10.0.0.0/16 le 32
end-set
prefix-set too-specific
  0.0.0.0/0 ge 25 le 32
end-set
route-policy example-one
  if destination in ten-net then
    drop
  else
    set local-preference 200
    apply set-comms
  endif
end-policy
route-policy set-comms
  set community (10:1234) additive
end-policy

route-policy example-three
  if destination in too-specific then
```
The following example displays information showing the usage and status of each policy that references the prefix set ten-net. The `brief` keyword limits the display to just a summary table and not the detailed information for the prefix set.

```
RP/0/RP0/CPU0:router# show rpl prefix-set ten-net references brief
```

**Usage**
- Direct -- Reference occurs in this policy
- Indirect -- Reference occurs via an apply statement

**Status**
- UNUSED -- Policy is not in use at an attachpoint (unattached)
- ACTIVE -- Policy is actively used at an attachpoint
- INACTIVE -- Policy is applied by an unattached policy

<table>
<thead>
<tr>
<th>Usage/Status</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>1</td>
</tr>
<tr>
<td>Indirect</td>
<td>1</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>0</td>
</tr>
<tr>
<td>INACTIVE</td>
<td>1</td>
</tr>
<tr>
<td>UNUSED</td>
<td>1</td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

**Table 116: show rpl prefix-set name references Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usage/Status</td>
<td>Displays the usage and status of all policies that reference the prefix set.</td>
</tr>
<tr>
<td>count</td>
<td>Number of policies that match each usage and status option.</td>
</tr>
</tbody>
</table>
show rpl rd-set

To display the configuration of route distinguisher (RD) sets, use the show rpl rd-set command in XR EXEC mode.

```
show rpl rd-set [ {name|states|brief} ]
```

**Syntax Description**

- **name** (Optional) Name of the RD set.
- **states** (Optional) Shows all unused, inactive, and active states.
- **brief** (Optional) Limits the display to a list of the names of all RD sets without their configurations.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

```
Modification               Release
This command was introduced. Release 6.0
```

**Usage Guidelines**

Because sets cannot hierarchically reference other sets or policies, no detail keyword exists as with the show rpl policy command.

**Task ID**

```
Task ID   Operations
route-policy  read
```

**Examples**

In the following example, the configuration of RD set rdset1 is displayed:

```
RP/0/RP0/CPU0:router# show rpl rd-set rdset1
rd-set rdset1
  10:151,
  100.100.100.1:153,
  100.100.100.62/31:63
end-set
```
show rpl rd-set attachpoints

To display all the policies used at an attach point that reference the named route distinguisher (RD) set, use the **show rpl rd-set attachpoints** command in XR EXEC mode.

```
show rpl rd-set name attachpoints
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of an RD set.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **show rpl rd-set attachpoints** command to display all the policies used at an attach point that reference the named RD set either directly or indirectly.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
route-policy rdsetmatch
  if rd in rdset1 then
    set community (10:112)
  elseif rd in rdset2 then
    set community (10:223)
  endif
end-policy

router bgp 10
  address-family vpnv4 unicast
  exit
  neighbor 10.0.101.1
  remote-as 11
  address-family vpnv4 unicast
  route-policy rdsetmatch in
!
```

Given this sample configuration, the **show rpl rd-set rdset1 attachpoints** command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl rd-set rdset1 attachpoints
BGP Attachpoint: Neighbor
Neighbor/Group type afi/safi in/out vrf name
```
This table describes the significant fields shown in the display.

**Table 117: show rpl rd-set attachpoints Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Neighbor/Group</td>
<td>BGP neighbor or neighbor group where the specified RD is used.</td>
</tr>
<tr>
<td>afi/safi</td>
<td>BGP address family or subaddress family where the RD set is used.</td>
</tr>
<tr>
<td>in/out</td>
<td>Direction</td>
</tr>
<tr>
<td>vrf name</td>
<td>VRF name where the RD set is used.</td>
</tr>
</tbody>
</table>
show rpl rd-set references

To list all the policies that reference the named route distinguisher (RD) set, use the `show rpl rd-set references` command in XR EXEC mode.

```
show rpl rd-set name references [brief]
```

**Syntax Description**

- **name**  Name of the RD set.
- **brief** (Optional) Limits the output to just a summary table and not the detailed information for the RD set.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl rd-set references` command to list all the policies that reference the named RD set.

Use the optional **brief** keyword to limit the output to just a summary table and not the detailed information for the named RD set.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
route-policy rdsetmatch
  if rd in rdset1 then
    set community (10:112)
  elseif rd in rdset2 then
    set community (10:223)
  endif
end-policy
!
router bgp 10
  address-family vpnv4 unicast
!
neighbor 10.0.101.1
  remote-as 11
  address-family vpnv4 unicast
  route-policy rdsetmatch in
!
```

Given this sample configuration, the `show rpl rd-set rdset1 references` command displays the following information:
RP/0/RP0/CPU0:router# show rpl rd-set rdset1 references

Usage Direct -- Reference occurs in this policy
Usage Indirect -- Reference occurs via an apply statement

Status UNUSED -- Policy is not in use at an attachpoint (unattached)
Status ACTIVE -- Policy is actively used at an attachpoint
Status INACTIVE -- Policy is applied by an unattached policy

<table>
<thead>
<tr>
<th>Usage/Status</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>1</td>
</tr>
<tr>
<td>Indirect</td>
<td>0</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>1</td>
</tr>
<tr>
<td>INACTIVE</td>
<td>0</td>
</tr>
<tr>
<td>UNUSED</td>
<td>0</td>
</tr>
</tbody>
</table>

route-policy      usage  policy status
--------------------- --------------------
rdsetmatch         Direct  ACTIVE

This table describes the significant fields shown in the display.

**Table 118: show rpl rd-set name references Field Descriptions**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>Name of the route policy.</td>
</tr>
<tr>
<td>usage</td>
<td>Type of reference usage for the route policy.</td>
</tr>
<tr>
<td>policy status</td>
<td>Status of the route policy.</td>
</tr>
</tbody>
</table>
show rpl route-policy

To display the configuration of route policies, use the `show rpl route-policy` command in XR EXEC mode.

```
show rpl route-policy [{name | detail|states|brief}]
```

### Syntax Description

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>(Optional) Name of a route policy.</td>
</tr>
<tr>
<td>detail</td>
<td>(Optional) Displays the configuration of all policies and sets that a policy uses.</td>
</tr>
<tr>
<td>states</td>
<td>(Optional) Shows all unused, inactive, and active states.</td>
</tr>
<tr>
<td>brief</td>
<td>(Optional) Limits the display to a list of the names of all extended community sets without their configurations.</td>
</tr>
</tbody>
</table>

### Command Default

No default behavior or values

### Command Modes

XR EXEC mode

### Command History

**Release** | **Modification**  
-------------|-------------------|
Release 6.0  | This command was introduced.

### Usage Guidelines

Use the optional `brief` keyword to limit the display to a list of the names of policies without their configurations.

### Examples

In the following example, the configuration of a route policy named policy_1 is displayed.

```
RP/0/RP0/CPU0:router# show rpl route-policy policy_1

route-policy policy_1
  if destination in prefix_set_1 and not destination in sample1 then
    if as-path in aspath_set_1 then
      set local-preference 300
      set origin igp
    elseif as-path in as_allowed then
      set local-preference 400
      set origin igp
    else
      set origin igp
      endif
    else
      drop
    endif
  set med 120
  set community (8660:612) additive
  apply set_lpref_from_comm
```
If the optional **detail** keyword is used, all routing policy language (RPL) policies and sets that route policy `policy_1` uses are displayed, as shown in the following example.

```plaintext
RP/0/RP0/CPU0:router# show rpl route-policy policy_1 detail

! prefix-set sample1
  0.0.0.0/0,
  0.0.0.0/0 ge 25 le 32,
  10.0.0.0/8 ge 8 le 32,
  192.168.0.0/16 ge 16 le 32,
  224.0.0.0/20 ge 20 le 32,
  240.0.0.0/20 ge 20 le 32
end-set
!
prefix-set prefix_set_1
  10.0.0.1/24 ge 24 le 32,
  10.0.0.5/24 ge 24 le 32,
  172.16.0.1/24 ge 24 le 32,
  172.16.5.5/24 ge 24 le 32,
  172.16.20.10/24 ge 24 le 32,
  172.30.0.1/24 ge 24 le 32,
  10.0.0.10/24 ge 24 le 32,
  172.18.0.1/24 ge 24 le 32,
  192.168.0.1/24 ge 24 le 32,
  192.168.20.10/24 ge 24 le 32,
  192.168.200.10/24 ge 24 le 32,
  192.168.255.254/24 ge 24 le 32
end-set
!
as-path-set as_allowed
  ios-regex '.+_1239_.*',
  ios-regex '.+_3561_.*',
  ios-regex '.+_701_.*',
  ios-regex '.+_666_.*',
  ios-regex '.+_1755_.*',
  ios-regex '.+_1756_.*'
end-set
!
as-path-set aspath_set_1
  ios-regex '_9148_',
  ios-regex '_5870_',
  ios-regex '_2408_',
  ios-regex '_2531_',
  ios-regex '_197_',
  ios-regex '_2992_'
end-set
!
route-policy set_lpref_from_comm
  if community matches-any (2:50) then
    set local-preference 50
  elseif community matches-any (2:60) then
    set local-preference 60
  elseif community matches-any (2:70) then
    set local-preference 70
  elseif community matches-any (2:80) then
    set local-preference 80
  else community matches-any (2:90) then
    set local-preference 90
endif
```
end-policy
!
route-policy policy_1
  if destination in prefix_set_1 and not destination in sample1 then
    if as-path in aspath_set_1 then
      set local-preference 300
      set origin igp
    elseif as-path in as_allowed then
      set local-preference 400
      set origin igp
    else
      set origin igp
    endif
  else
    drop
  endif
set med 120
set community (8660:612) additive
apply set_lpref_from_comm
end-policy
show rpl route-policy attachpoints

To display all the policies used at an attach point that reference the named policy, use the `show rpl route-policy attachpoints` command in XR EXEC mode.

show rpl route-policy name attachpoints

**Syntax Description**

| name | Name of a policy. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl route-policy attachpoints` command to display all the policies used at an attach point that reference the named policy either directly or indirectly.

The policy name is required.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
    address-family ipv4 unicast
    !
    neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
    route-policy policy_1 in
    !
    neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
    route-policy policy_2 in
    !

RP/0/RP0/CPU0:router# show rpl route-policy policy_1

route-policy policy_1
    if (destination in prefix_set_ex1) then
        set local-preference 100
    endif
```
show rpl route-policy policy_2

route-policy policy_2
  if (destination in prefix_set_ex1) then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy
!

The following command displays the route policy attach points for policy_2:

RP/0/RP0/CPU0:router# show rpl route-policy policy_2 attachpoints

BGP Attachpoint: Neighbor

<table>
<thead>
<tr>
<th>Neighbor/Group</th>
<th>type</th>
<th>afi/safi</th>
<th>in</th>
<th>out</th>
<th>vrf name</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.101.2</td>
<td>--</td>
<td>IPv4/uni</td>
<td>in</td>
<td></td>
<td>default</td>
</tr>
<tr>
<td>10.0.101.2</td>
<td>--</td>
<td>IPv4/uni</td>
<td>out</td>
<td></td>
<td>default</td>
</tr>
</tbody>
</table>

This table describes the significant fields shown in the display.

Table 119: show rpl route-policy attachpoints Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BGP Attachpoint</td>
<td>Location of the attach point.</td>
</tr>
<tr>
<td>Neighbor/Group</td>
<td>IP address of the attach point on the neighbor.</td>
</tr>
<tr>
<td>type</td>
<td>Displays the address family mode.</td>
</tr>
<tr>
<td>afi/safi</td>
<td>Address family identifier or subsequent address family identifier.</td>
</tr>
<tr>
<td>vrf name</td>
<td>Name of the VPN routing and forwarding (VRF) instance.</td>
</tr>
</tbody>
</table>
show rpl route-policy inline

To display all policies and sets that a policy uses expanded inline, use the `show rpl route-policy inline` command in XR EXEC mode.

```
show rpl route-policy name inline
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>Name of a policy.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl route-policy inline` command to examine the configuration of a specified route policy. All policies and sets that a policy uses are gathered together and displayed expanded inline.

The policy name is required.

**Examples**

The following command displays the route policy `policy_1`:

```
RP/0/RP0/CPU0:router# show rpl route-policy policy_1

! route-policy policy_1
  if destination in prefix_set_1 and not destination in martians then
    if as-path in aspath_set_1 then
      set local-preference 300
      set origin igp
    elseif as-path in as_allowed then
      set local-preference 400
      set origin igp
    else
      set origin igp
    endif
  else
    drop
  endif
  set med 120
  set community (8660:612) additive
  apply set_lpref_from_comm
end-policy
```
The following command displays the route policy policy_1 and all the other sets or policies it refers too inline. Adding the inline keyword causes the configuration to be displayed inline for all RPL objects that the route-policy policy_1 uses.

```
RP/0/RP0/CPU0:router#show rpl policy policy_1 inline

route-policy policy_1
  if destination in (91.5.152.0/24 ge 24 le 32, 91.220.152.0/24 ge 24 le 32, 61.106.52.0/24 ge 24 le 32, 222.168.199.0/24 ge 24 le 32, 93.76.114.0/24 ge 24 le 32, 41.195.116.0/24 ge 24 le 32, 35.92.152.0/24 ge 24 le 32, 143.144.96.0/24 ge 24 le 32, 79.218.81.0/24 ge 24 le 32, 75.213.219.0/24 ge 24 le 32, 178.220.61.0/24 ge 24 le 32, 27.195.65.0/24 ge 24 le 32, 79.218.81.0/24 ge 24 le 32, 75.213.219.0/24 ge 24 le 32, 178.220.61.0/24 ge 24 le 32, 27.195.65.0/24 ge 24 le 32, 79.218.81.0/24 ge 24 le 32, 75.213.219.0/24 ge 24 le 32, 178.220.61.0/24 ge 24 le 32, 27.195.65.0/24 ge 24 le 32) and not destination in (0.0.0.0/0, 0.0.0.0/0 ge 25 le 32, 10.0.0.0/8 ge 8 le 32, 192.168.0.0/16 ge 16 le 32, 224.0.0.0/20 ge 20 le 32, 240.0.0.0/20 ge 20 le 32) then
    if as-path in (ios-regex '_9148_', ios-regex '_5870_', ios-regex '_2408_', ios-regex '_2531_', ios-regex '_197_', ios-regex '_2992_') then
      set local-preference 300
      set origin igp
      elseif as-path in (ios-regex '.* _1239_ .*', ios-regex '.* _3561_ .*', ios-regex '.* _701_ .*', ios-regex '.* _666_ .*', ios-regex '.* _1755_ .*', ios-regex '.* _1756_ .*') then
        set local-preference 400
        set origin igp
      else
        set origin igp
      endif
    endif
  else
    drop
  endif
set med 120
set community (8660:612) additive
  # apply set_lpref_from_comm
  if community matches-any (2:50) then
    set local-preference 50
  elseif community matches-any (2:60) then
    set local-preference 60
  elseif community matches-any (2:70) then
    set local-preference 70
  elseif community matches-any (2:80) then
    set local-preference 80
  elseif community matches-any (2:90) then
    set local-preference 90
  endif
  # end-apply set_lpref_from_comm
end-policy
```
show rpl route-policy references

To list all the policies that reference the named policy, use the `show rpl route-policy references` command in XR EXEC mode.

```
show rpl route-policy name references [brief]
```

**Syntax Description**
- `name` Name of a prefix set.
- `brief` (Optional) Limits the output to just a summary table and not the detailed information for the named policy.

**Command Default**
No default behavior or values

**Command Modes**
XR EXEC mode

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Use the `show rpl route-policy references` command to list all the policies that reference the named policy.

Use the optional `brief` keyword to limit the output to just a summary table and not the detailed information for the policy.

**Examples**
This example shows the following sample configuration:

```
prefix-set ten-net
  10.0.0.0/16 le 32
end-set
prefix-set too-specific
  0.0.0.0/0 ge 25 le 32
end-set
route-policy example-one
  if destination in ten-net then
    drop
  else
    set local-preference 200
    apply set-comms
  endif
end-policy
route-policy set-comms
  set community (10:1234) additive
end-policy
route-policy example-three
  if destination in too-specific then
    drop
```
else
  apply example-one
  pass
endif
end-policy

The following command displays information about the policy set-comms and how it is referenced:

RP/0/RP0/CPU0:router# show rpl route-policy set-comms references

Usage Direct -- Reference occurs in this policy
Usage Indirect -- Reference occurs via an apply statement
Status UNUSED -- Policy is not in use at an attachpoint (unattached)
Status ACTIVE -- Policy is actively used at an attachpoint
Status INACTIVE -- Policy is applied by an unattached policy

<table>
<thead>
<tr>
<th>Usage/Status</th>
<th>count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct</td>
<td>1</td>
</tr>
<tr>
<td>Indirect</td>
<td>1</td>
</tr>
<tr>
<td>ACTIVE</td>
<td>0</td>
</tr>
<tr>
<td>INACTIVE</td>
<td>1</td>
</tr>
<tr>
<td>UNUSED</td>
<td>1</td>
</tr>
</tbody>
</table>

route-policy usage policy status
--------------------------------------------------------------
example-one Direct INACTIVE
example-three Indirect UNUSED

The direct usage indicates that the route policy example-one directly applies the policy set-comms, that is, example-one has a line in the form apply set-comms. The usage Indirect indicates that the route policy example-three does not directly apply the route policy set-comms. However, the route policy example-three does apply the policy example-one, which in turn applies the policy set-comms, so there is an indirect reference from example-three to the route policy set-comms.

The status column indicates one of three states. A policy is active if it is in use at an attach point. In the example provided, neither example-one nor example-three is in use at an attach point, which leaves two possible states: UNUSED or INACTIVE. The route policy example-one is inactive because it has some other policy (example-three) that references it, but neither example-one nor any of the policies that reference it (example-one) are in use at an attach point. The route policy example-three has a status of unused because it is not used at an attach point and no other route policies in the system refer to it.

This table describes the significant fields shown in the display.

### Table 120: show rpl route-policy references Field Descriptions

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usage/Status</td>
<td>Displays the usage and status of all policies that reference the specified policy.</td>
</tr>
<tr>
<td>count</td>
<td>Number of policies that match each usage and status option.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>route-policy</td>
<td>One name for multiple policies that reference the specified policy.</td>
</tr>
<tr>
<td>usage</td>
<td>Type of usage for the policy.</td>
</tr>
<tr>
<td>policy status</td>
<td>Status of the policy.</td>
</tr>
</tbody>
</table>
show rpl route-policy uses

To display information about a specified named policy, use the `show rpl route-policy uses` command in XR EXEC mode.

```
show rpl route-policy name uses {policies|sets|all} [direct]
```

**Syntax Description**

- **name** Name of a policy.
- **policies** Generates a list of all policies that the named policy uses.
- **sets** Lists all named sets that are used by the policy.
- **all** Generates a list of both sets and policies that the named policy references.
- **direct** (Optional) Lists only the policies or sets used directly in the named policy block. Set or policy references that occur as a result of an `apply` statement are not listed.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl route-policy uses` command to display information about a specified named policy.

**Examples**

This example shows the following sample configuration:

```
prefix-set ten-net
  10.0.0.0/16 le 32
end-set
prefix-set too-specific
  0.0.0.0/0 ge 25 le 32
end-set
route-policy example-one
  if destination in ten-net then
    drop
  else
    set local-preference 200
    apply set-comms
  endif
end-policy
route-policy set-comms
  set community (10:1234) additive
end-policy
```
route-policy example-three
if destination in too-specific then
drop
else
apply example-one
pass
endif
end-policy

The following command lists the policies one and set-comms. It also lists the prefix sets too-specific and ten-net.

RP/0/RP0/CPU0:router# show rpl route-policy example-three uses all

Policies directly and indirectly applied by this policy:
----------------------------------------------------------
example-one set-comms

Sets referenced directly and indirectly (via applied policies) in this policy:
----------------------------------------
type prefix-set:
  ten-net too-specific

The sets example-one and set-comms are listed as policies that are used by the policy example-three. The policy example-one is listed because route policy example-three uses it in an apply statement. The policy set-comms is also listed because example-one applies it. Similarly, the prefix-set too-specific is used directly in the if statement in the policy example-three, and the prefix-set ten-net is used in the policy example-one. The optional direct keyword can be used to limit the output to just those sets and policies that are used within the example-three block itself, as shown in the following example:

RP/0/RP0/CPU0:router# show rpl route-policy example-three uses all direct

Policies directly applied by this policy:
------------------------------------------
example-one

Sets used directly in this policy
------------------------------------
type prefix-set:
  ten-net too-specific

As can be seen in the output, the route policy set-comms and the prefix set ten-net are no longer included in the output when the direct keyword is used. The direct form of the command considers only those sets or policies used in the specified route policy and any additional policies or sets that may be used if you follow the hierarchy of apply statements.
This table describes the significant fields shown in the display.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>Displays the type used in the policy configuration.</td>
</tr>
<tr>
<td></td>
<td>Values for type are prefix-set, community-set, extcommunity-set, and as-path-set.</td>
</tr>
</tbody>
</table>
show rpl unused as-path-set

To display the AS path sets that are defined but not used by a policy at an attach point or referenced in a policy using an apply statement, use the show rpl unused as-path-set command in XR EXEC mode.

```
show rpl unused as-path-set [detail]
```

**Syntax Description**
- **detail** (Optional) Displays the content of the object and all referenced objects for unused AS path sets.

**Command Default**
No default behavior or values

**Command Modes**
XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the show rpl unused as-path-set command to display all AS path sets that are not used in a policy at an attach point either directly or indirectly and are not referenced by any policies in the system.

**Task ID**
- read

**Examples**

This example shows the following sample configuration:

```
routing bgp 2
  address-family ipv4 unicast
  !
  neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
      route-policy policy_1 in
      !
  neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
      route-policy policy_2 in
      !
  !
as-path-set as_path_set_ex1
  ios-regex '^_65500_$',
  ios-regex '^_65501_$'
  end-set
  !
as-path-set as_path_set_ex2
  ios-regex '^_65502_$',
  ios-regex '^_65503_$'
  end-set
  !
```
as-path-set as_path_set_ex3
  ios-regex '^_65504_\$',
  ios-regex '^_65505_\$
end-set
!
route-policy sample
  if (destination in sample) then
    drop
  endif
end-policy
!
route-policy policy_1
  if (destination in prefix_set_ex1) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex1) then
    set community (10:333) additive
  endif
end-policy
!
route-policy policy_2
  if (destination in prefix_set_ex1) then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy
!
route-policy policy_3
  if (destination in prefix_set_ex2) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex2) then
    set community (10:333) additive
  endif
end-policy
!
route-policy policy_4
  if (destination in prefix_set_ex2) then
    if (community matches-any comm_set_ex2) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex2) then
      set community (10:999) additive
    endif
  endif
end-policy
!
route-policy policy_5
  apply sample
  apply policy_3
end-policy

Given this sample configuration, the show rpl unused as-path-set command displays the following information:

RP/0/RP0/CPU0:router# show rpl unused as-path-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following as-path-sets are UNUSED
-------------------------------------
as_path_set_ex3
show rpl unused community-set

To display the community sets that are defined but not used by a policy at an attach point or referenced in a policy using an apply statement, use the show rpl unused community-set command in XR EXEC mode.

```
show rpl unused community-set [detail]
```

**Syntax Description**

- **detail** (Optional) Displays the content of the object and all referenced objects for unused community sets.

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the show rpl unused community-set command to display all the community sets that are not used in a policy at an attach point either directly or indirectly and are not referenced by any policies in the system.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast
  !
  neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
      route-policy policy_1 in
      !
  neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
      route-policy policy_2 in
      !

  community-set comm_set_ex1
    65500:1,
    65500:2,
    65500:3
  end-set
  !
  community-set comm_set_ex2
    65501:1,
    65501:2,
```
Given this sample configuration, the `show rpl unused community-set` command displays the following information:
show rpl unused community-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following community-sets are UNUSED
---------------------------------------
comm_set_ex3
**show rpl unused extcommunity-set**

To display the extended community sets that are defined but not used by a policy at an attach point or referenced in a policy using an `apply` statement, use the `show rpl unused extcommunity-set` command in XR EXEC mode.

```
show rpl unused extcommunity-set [{cost|detail|rt|soo}]
```

<table>
<thead>
<tr>
<th>Syntax Description</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cost</td>
<td>(Optional) Displays the unused extended-community cost objects.</td>
</tr>
<tr>
<td>rt</td>
<td>(Optional) Displays the unused extended community RT objects.</td>
</tr>
<tr>
<td>soo</td>
<td>(Optional) Displays the unused extended-community SoO objects.</td>
</tr>
<tr>
<td>detail</td>
<td>(Optional) Displays the content of the object and all referenced objects for unused extended community sets.</td>
</tr>
</tbody>
</table>

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `show rpl unused extcommunity-set` command to display all extended community sets that are not used in a policy at an attach point either directly or indirectly and are not referenced by any policies in the system.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

The following is sample output for the `show rpl unused extcommunity-set` command:

```
RP/0/RP0/CPU0:router# show rpl unused extcommunity-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following extcommunity-sets are UNUSED
------------------------------------------
ext_comm_set_ex3
```
show rpl unused prefix-set

To display the prefix sets that are defined but not used by a policy at an attach point or referenced in a policy using an apply statement, use the show rpl unused prefix-set command in XR EXEC mode.

```
show rpl unused prefix-set [detail]
```

**Syntax Description**

| Syntax Description       | detail (Optional) Displays the content of the object and all referenced objects for unused prefix sets. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the show rpl unused prefix-set command to display all prefix sets that are not used in a policy at an attach point either directly or indirectly and are not referenced by any policies in the system.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read</td>
</tr>
</tbody>
</table>

**Examples**

This example shows the following sample configuration:

```
router bgp 2
  address-family ipv4 unicast
  !
  neighbor 10.0.101.2
    remote-as 100
    address-family ipv4 unicast
      route-policy policy_1 in
      !
  !
  neighbor 10.0.101.3
    remote-as 12
    address-family ipv4 unicast
      route-policy policy_2 in
      !
  !

prefix-set sample
  0.0.0.0/0,
  0.0.0.0/0 ge 25 le 32,
  10.0.0.0/8 ge 8 le 32,
  192.168.0.0/16 ge 16 le 32,
  224.0.0.0/20 ge 20 le 32,
  240.0.0.0/20 ge 20 le 32
end-set
```
prefix-set prefix_set_ex1
  10.0.0.0/16 ge 16 le 32,
  0.0.0.0/0 ge 25 le 32,
  0.0.0.0/0
end-set
!
prefix-set prefix_set_ex2
  220.220.220.0/24 ge 24 le 32,
  220.220.120.0/24 ge 24 le 32,
  220.220.130.0/24 ge 24 le 32
end-set
!
prefix-set prefix_set_ex3
  221.221.220.0/24 ge 24 le 32,
  221.221.120.0/24 ge 24 le 32,
  221.221.130.0/24 ge 24 le 32
end-set
!
route-policy sample
  if (destination in sample) then
    drop
  endif
end-policy
!
route-policy policy_1
  if (destination in prefix_set_ex1) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex1) then
    set community (10:333) additive
  endif
end-policy
!
route-policy policy_2
  if (destination in prefix_set_ex1) then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy
!
route-policy policy_3
  if (destination in prefix_set_ex2) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex2) then
    set community (10:333) additive
  endif
end-policy
!
route-policy policy_4
  if (destination in prefix_set_ex2) then
    if (community matches-any comm_set_ex2) then
      set community (10:666) additive
    endif
    if (extcommunity matches-any ext_comm_set_rt_ex2) then
      set community (10:999) additive
    endif
  endif
end-policy
!
route-policy policy_5
   apply sample
   apply policy_3
end-policy
--------------------------
ext_comm_set_ex3

Given this sample configuration, the `show rpl unused prefix-set` command displays the following information:

```
RP/0/RP0/CPU0:router# show rpl unused prefix-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following prefix-sets are UNUSED
------------------------------------
prefix_set_ex3
```
show rpl unused rd-set

To display the route distinguisher (RD) sets that are defined but not used by a policy at an attach point or referenced in a policy using an apply statement, use the show rpl unused rd-set command in XR EXEC mode.

show rpl unused rd-set [detail]

Syntax Description

detail (Optional) Displays the content of the object and all referenced objects for unused RD sets.

Command Default

No default behavior or values

Command Modes

XR EXEC mode

Command History

Release 6.0 This command was introduced.

Usage Guidelines

Use the show rpl unused rd-set command to display all of the RD sets that are not used in a policy at an attach point either directly or indirectly and are not referenced by any policies in the system.

Task ID

Task ID Operations
route-policy read

Examples

The show rpl unused rd-set command displays the following information:

RP/0/RP0/CPU0:router# show rpl unused rd-set

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following rd-sets are UNUSED
---------------------------------------------------------------
None found with this status.
show rpl unused route-policy

To display the route policies that are defined but not used at an attach point or referenced using an apply statement, use the **show rpl unused route-policy** command in XR EXEC mode.

```
show rpl unused route-policy [detail]
```

**Syntax Description**
- **detail** (Optional) Displays the content of the object and all referenced objects for unused route policies.

**Command Default**
No default behavior or values

**Command Modes**
XR EXEC mode

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Use the **show rpl unused route-policy** command to display route policies that are defined but not used at an attach point or referenced from another policy using an apply statement.

**Task ID**
- **route-policy** read

**Examples**
This example shows the following sample configuration:

```
RP/0/RP0/CPU0:router# show run | begin prefix-set

Building configuration...
prefix-set prefix_set_ex1
  10.0.0.0/16 ge 16 le 32,
  0.0.0.0/0 ge 25 le 32,
  0.0.0.0/0
end-set

prefix-set prefix_set_ex2
  220.220.220.0/24 ge 24 le 32,
  220.220.120.0/24 ge 24 le 32,
  220.220.130.0/24 ge 24 le 32
end-set

as-path-set as_path_set_ex1
  ios-regex '^_65500_$',
  ios-regex '^_65501_$'
end-set

as-path-set as_path_set_ex2
  ios-regex '^_65502_$',
  ios-regex '^_65503_$'
end-set

as-path-set as_path_set_ex3
```
ios-regex '^_65504_$',
ios-regex '^_65505_$'
end-set
!
community-set comm_set_ex1
  65500:1,
  65500:2,
  65500:3
end-set
!
community-set comm_set_ex2
  65501:1,
  65501:2,
  65501:3
end-set
!
exitcommunity-set rt ext_comm_set_rt_ex1
  1.2.3.4:34
end-set
!
exitcommunity-set rt ext_comm_set_rt_ex2
  2.3.4.5:36
end-set
!
route-policy sample
  if (destination in sample) then
    drop
  endif
end-policy
!
route-policy policy_1
  if (destination in prefix_set_ex1) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex1) then
    set community (10:333) additive
  endif
end-policy
!
route-policy policy_2
  if (destination in prefix_set_ex1) then
    if (community matches-any comm_set_ex1) then
      set community (10:666) additive
    endif
    if (exitcommunity rt matches-any ext_comm_set_rt_ex1) then
      set community (10:999) additive
    endif
  endif
end-policy
!
route-policy policy_3
  if (destination in prefix_set_ex2) then
    set local-preference 100
  endif
  if (as-path in as_path_set_ex2) then
    set community (10:333) additive
  endif
end-policy
!
route-policy policy_4
  if (destination in prefix_set_ex2) then
    if (community matches-any comm_set_ex2) then
      set community (10:666) additive
    endif
  endif

if (extcommunity rt matches-any ext_comm_set_rt_ex2) then
    set community (10:999) additive
endif
endif
end-policy
!
route-policy policy_5
    apply sample
    apply policy_3
end-policy
!
route ipv4 0.0.0.0/0 10.91.37.129
route ipv4 10.91.36.0/23 10.91.37.129
route ipv4 10.91.38.0/24 10.91.37.129
end

In the following example, route policies that are defined but not used at an attach point or referenced from another policy using an `apply` statement are displayed using the `show rpl unused route-policy` command.

RP/0/RP0/CPU0:router# show rpl unused route-policy

ACTIVE -- Referenced by at least one policy which is attached
INACTIVE -- Only referenced by policies which are not attached
UNUSED -- Not attached (directly or indirectly) and not referenced

The following policies are (UNUSED)

--------------------
policy_1
policy_2
policy_4
policy_5
# source in

To test the source of a Border Gateway Protocol (BGP) route against the address contained in either a named or an inline prefix set, use the **source in** command in route-policy configuration mode.

```
source in \{prefix-set-name|inline-prefix-set|parameter\}
```

## Syntax Description

- **prefix-set-name**: Name of a prefix set.
- **inline-prefix-set**: Inline prefix set. The inline prefix set must be enclosed in parentheses.
- **parameter**: Parameter name. The parameter name must be preceded with a “$.”

### Command Default

No default behavior or values

### Command Modes

Route-policy configuration

### Command History

**Release** | **Modification**
--- | ---
**Release 6.0** | This command was introduced.

### Usage Guidelines

Use the **source in** command as a conditional expression within an **if** statement to test the source of the route against the data in either a named or an inline prefix set. A comparison that references a prefix set with zero elements in it returns false.

- **Note**

For a list of all conditional expressions available within an **if** statement, see the **if** command.

The source of a BGP route is the IP peering address of the neighboring router from which the route was received.

The prefix set can contain both IPv4 and IPv6 prefix specifications.

### Examples

In the following example, the source of a BGP route is tested against the data in the prefix set **my-prefix-set**:

```
RP/0/RP0/CPU0:router(config)# route-policy policy-A
RP/0/RP0/CPU0:router(config-rpl)# if source in my-prefix-set then
```

In this example, the source of a BGP route is tested against the data in an inline IPv4 prefix set:

```
RP/0/RP0/CPU0:router(config)# route-policy policy-B
```
In this example, the source of a route is tested against the data in an inline IPv6 prefix set:

```
RP/0/RP0/CPU0:router(config-rpl)# if source in (10.0.0.8, 10.0.0.20) then

RP/0/RP0/CPU0:router(config)# route-policy policy-C
RP/0/RP0/CPU0:router(config-rpl)# if source in (2001:0:0:1::/64, 2001:0:0:2::/64) then
```
suppress-route

To indicate that a given component of a BGP aggregate should be suppressed, use the suppress-route command in route-policy configuration mode.

**suppress-route**

**Syntax Description**
This command has no arguments or keywords.

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the suppress-route command to indicate that a given component of an aggregate should be suppressed, that is, not advertised by BGP. See the unsuppress-route, on page 780 command for information on overriding the suppress-route command for individual neighbors.

The suppress-route command can be used as an action statement within an if statement. For a list of all action statements available within an if statement, see the if command.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>route-policy</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

In the following example, if the destination is in 10.1.0.0/16, then the route is not advertised:

```
RP/0/RP0/CPU0:router(config)# route-policy check-aggregate
RP/0/RP0/CPU0:router(config-rpl)# if destination in (10.1.0.0/16) then
RP/0/RP0/CPU0:router(config-rpl-if)# suppress-route
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl-if)# end-policy
```
tag

To match a specific tag value, use the `tag` command in route-policy configuration mode.

```
tag {eq|ge|le|is} {integerparameter}
```

**Syntax Description**

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>eq</td>
<td>Equal to</td>
</tr>
<tr>
<td>ge</td>
<td>Greater than or equal to</td>
</tr>
<tr>
<td>le</td>
<td>Less than or equal to</td>
</tr>
<tr>
<td>is</td>
<td>Is</td>
</tr>
</tbody>
</table>

- **integer** Integer value. Range is 0 to 4294967295.
- **parameter** Parameter name. The parameter name must be preceded with a “$.”

**Command Default**
No default behavior or values

**Command Modes**
Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced</td>
</tr>
</tbody>
</table>

**Usage Guidelines**
Use the `tag` command as a conditional expression within an `if` statement to match a specific tag value.

**Note**
For a list of all conditional expressions available within an `if` statement, see the `if` command.

A tag is a 32-bit integer that can be associated with a given route within the RIB.

The `eq` operator matches either a specific tag value or a parameter value. Its variants `ge` and `le` match a range of tag values that are either greater than or equal to or less than or equal to the supplied value or parameter.

**Examples**
In the following example, if the tag equals 10, then the condition returns true:

```
RP/0/RSP0RP0/CPU0:router(config-rpl)# if tag eq 10 then
```
tag in

To match a tag entry in a named tag set or inline tag set, use the **tag in** command in route-policy configuration mode.

**tag in** \{**tag-set-name** inline-tag-set \} parameter

**Syntax Description**

- **tag-set-name** Name of a tag set. The tag-set accepts 32-bit Integer value. Range is 0 to 4294967295.
- **inline-tag-set** Inline tag set. The inline tag set must be enclosed in parentheses.
- **parameter** Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Release 6.0.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the **tag in** command as a conditional expression within an **if** statement to match a tag entry in a named tag set or inline tag set.

**Note**

For a list of all conditional expressions available within an **if** statement, see the **if** command.

This command takes either a named tag set or an inline tag set value as an argument. The condition returns true if the tag entry matches any entry in the tag set or inline tag set. An attempt to match a tag using a tag set that is defined but contains no elements returns false.

The routing policy language (RPL) provides the ability to test tags for a match to a list of tag match specifications using the **in** operator. The **tag in** command is protocol-independent.

**Examples**

In the following example, a tag set named **my-tag-set** is defined and a route policy named **use-tag-in** is created. Within the **use-tag-in** route policy, the **tag in** command is used within an **if** statement to learn if the tag is in the tag-set named **my-tag-set**. If it is, then local preference is set to 100. If it is not in **my-tag-set** but does match the next tag specifications, then local preference is set to 200.

```
RP/0/RP0/CPU0:router(config)#tag-set my-tag-set
RP/0/RP0/CPU0:router(config-tag)#1000
```
RP/0/RP0/CPU0:router(config-tag)#3000
RP/0/RP0/CPU0:router(config-tag)#end-set

RP/0/RP0/CPU0:router(config)#route-policy use-tag-in
RP/0/RP0/CPU0:router(config-rpl)#if tag in my-tag-set then
RP/0/RP0/CPU0:router(config-rpl-if)#set local-preference 100
RP/0/RP0/CPU0:router(config-rpl-if)#elseif tag in (2000, 4000) then
RP/0/RP0/CPU0:router(config-rpl-elseif)#set local-preference 200
RP/0/RP0/CPU0:router(config-rpl-elseif)#endif
RP/0/RP0/CPU0:router(config-rpl)#end policy
tag-set

To enter tag set configuration mode and define a tag set, use the `tag-set` command in XR Config mode. To remove a named tag set, use the `no` form of this command.

```
tag-set name
no tag-set name
```

**Syntax Description**

- **name**
  - Name of a tag set.

**Command Default**

None

**Command Modes**

XR Config

**Command History**

- **Release**
  - Release 6.0.0
- **Modification**
  - This command was introduced.

**Usage Guidelines**

Use the `tag-set` command to enter tag set configuration mode and define a tag set. A tag-set is a 32-bit integer that can be associated with a given route within the RIB.

**Task ID**

- **Task ID**
  - route-policy
  - read, write

**Examples**

In the following example, a tag set named `my-tag-set` is defined and a route policy named `use-tag-in` is created. Within the `use-tag-in` route policy, the `tag in` command is used within an `if` statement to learn if the tag is in the tag-set named `my-tag-set`. If it is, then local preference is set to 100. If it is not in `my-tag-set` but does match the next tag specifications, then local preference is set to 200.

```
RP/0/RP0/CPU0:router (config) # tag-set my-tag-set
RP/0/RP0/CPU0:router (config-tag) # 1000
RP/0/RP0/CPU0:router (config-tag) # 3000
RP/0/RP0/CPU0:router (config-tag) # end-set

RP/0/RP0/CPU0:router (config) # route-policy use-tag-in
RP/0/RP0/CPU0:router (config-rpl) # if tag in my-tag-set then
  RP/0/RP0/CPU0:router (config-rpl-if) # set local-preference 100
  RP/0/RP0/CPU0:router (config-rpl-if) # elseif tag in (2000, 4000) then
  RP/0/RP0/CPU0:router (config-rpl-elseif) # set local-preference 200
  RP/0/RP0/CPU0:router (config-rpl-elseif) # endif
RP/0/RP0/CPU0:router (config-rpl) # end policy
```
unsuppress-route

To indicate that a given component of a BGP aggregate should be unsuppressed, use the `unsuppress-route` command in route-policy configuration mode.

**unsuppress-route**

**Syntax Description**

This command has no arguments or keywords.

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `unsuppress-route` command to indicate that a given component of an aggregate should be unsuppressed, that is, allowed to be advertised by BGP again. This command affects routes that have been suppressed in the generation of BGP aggregates. If the request to unsuppress a route is encountered in a policy at a neighbor-out attach point, it guarantees that the routes that it affects are advertised to that neighbor even if that route was suppressed using the `suppress-route` command in a policy at the aggregation attach point.

The `unsuppress-route` command can be used as an action statement within an `if` statement. For a list of all action statements available within an `if` statement, see the `if` command.

**Examples**

In the following example, if the destination is in 10.1.0.0/16, then the route is not advertised:

```plaintext
RP/0/RP0/CPU0:router(config)# route-policy check-aggregate
RP/0/RP0/CPU0:router(config-rpl)# if destination in (10.1.0.0/16) then
RP/0/RP0/CPU0:router(config-rpl-if)# unsuppress-route
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)# end-policy
```

Assuming that the policy is attached at a neighbor-out attach point, if the route 10.1.0.0/16 was suppressed in a policy at an aggregation attach point, 10.1.0.0/16 is advertised to the neighbor. Routes continue to be suppressed in advertisements to other BGP neighbors unless a specific policy is attached to unsuppress the route.
**vpn-distinguisher is**

To match a specific Border Gateway Protocol (BGP) VPN distinguisher, use the `vpn-distinguisher is` command in route-policy configuration mode.

```
vpn-distinguisher is {numberparameter}
```

**Syntax Description**

- `number` Value assigned to a 32-bit unsigned integer. Range is from 1 to 4294967295.
- `parameter` Parameter name. The parameter name must be preceded with a “$.”

**Command Default**

No default behavior or values

**Command Modes**

Route-policy configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

Use the `vpn-distinguisher is` command as a conditional expression within an `if` statement to test the value of the origin attribute.

A VPN distinguisher is used in Layer 3 VPN networks for enhanced individual VPN control and to avoid route target mapping at AS boundaries in inter-AS VPN networks. Route target extended communities are removed at neighbor outbound and the VPN distinguisher value is applied on the BGP route as an extended community. When the route is received on a neighboring router in another AS, the VPN distinguisher is removed and mapped to a route target extended community.

For a list of all conditional expressions available within an `if` statement, see the `if` command.

This command can be parameterized.

**Examples**

In the following example, the origin is tested within an `if` statement to learn if it is either `igp` or `egp`:

```
RP/0/RP0/CPU0:router(config-rpl)# if origin is igp or origin is egp then
```

In the following example, a parameter is used to match a specific origin type:

```
RP/0/RP0/CPU0:router(config)# route-policy bar($origin)
```
vpn-distinguisher is

RP/0/RP0/CPU0:router(config-rpl)# if origin is $origin then
RP/0/RP0/CPU0:router(config-rpl-if)# set med 20
RP/0/RP0/CPU0:router(config-rpl-if)# endif
RP/0/RP0/CPU0:router(config-rpl)#
Static Routing Commands

This module describes the commands used to establish static routes on Cisco NCS 5000 Series Routers.

For detailed information about static routing concepts, configuration tasks, and examples, see the Implementing Static Routes on Cisco NCS 5000 Series Routers module in the Routing Configuration Guide for Cisco NCS 5000 Series Routers.

Currently, only default VRF is supported. VPNv4, VPNv6 and VPN routing and forwarding (VRF) address families will be supported in a future release.

- address-family (static), on page 784
- maximum path (static), on page 785
- route (static), on page 786
- router static, on page 788
address-family (static)

To enter various address family configuration modes while configuring static routes, use the `address-family` command in the appropriate configuration mode. To disable support for an address family, use the `no` form of this command.

```
address-family {ipv4|ipv6} {unicast}
no address-family {ipv4|ipv6} {unicast}
```

**Syntax Description**
- `ipv4` Specifies IP Version 4 address prefixes.
- `ipv6` Specifies IP Version 6 address prefixes. This option is available only in static router configuration mode.
- `unicast` Specifies unicast address prefixes.

**Command Default**
All static routes belong to the default VRF as static routing does not support VRFs other than default VRF.

**Command Modes**
- Router static configuration
- VRF router static configuration

**Command History**
- **Release 6.0** This command was introduced.

**Usage Guidelines**
Use the `address-family` command to enter various address family configuration modes while configuring static routing sessions. From address family configuration mode, you can configure static routes using the `route` command.

**Examples**
The following example shows how to enter IPv6 unicast address family mode:

```
RP/0/RP0/CPU0:router(config)# router static
RP/0/RP0/CPU0:router(config-static)# address-family ipv6 unicast
RP/0/RP0/CPU0:router(config-static-afi)#
```
maximum path (static)

To change the maximum number of allowable static routes, use the maximum path command in static router configuration mode. To remove the maximum path command from the configuration file and restore the system to its default condition, use the no form of this command.

maximum path {ipv4|ipv6} value
no maximum path {ipv4|ipv6} value

**Syntax Description**

<table>
<thead>
<tr>
<th>Syntax</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ipv4</td>
<td>Specifies IP Version 4 (IPv4) or IP Version 6 (IPv6) address prefixes.</td>
</tr>
<tr>
<td>value</td>
<td>Maximum number of static routes for the given AFI. The range is 1 to 140000.</td>
</tr>
</tbody>
</table>

**Command Default**

value : 4000

**Command Modes**

Static router configuration

**Command History**

<table>
<thead>
<tr>
<th>Release</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0</td>
<td>This command was introduced.</td>
</tr>
</tbody>
</table>

**Usage Guidelines**

If you use the maximum path command to reduce the configured maximum allowed number of static routes for a given table below the number of static routes currently configured, the change is rejected. In addition, if you commit a batch of routes that would, when grouped, push the number of static routes configured above the maximum allowed, the first $n$ routes in the batch and the number previously configured are accepted, and the remainder rejected. The $n$ argument is the difference between the maximum number allowed and the number previously configured.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>static</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to set the maximum number of static IPv4 routes to 100000:

```
RP/0/RP0/CPU0:router(config-static)# maximum path ipv4 100000
```

The following example shows how to remove the preceding configuration and set the maximum number of static IPv4 routes back to the default:

```
RP/0/RP0/CPU0:router(config-static)# no maximum path ipv4 100000
```
route (static)

To establish static routes, use the `route` command in address family configuration mode. To remove the `route` command from the configuration, use the `no` form of this command.

```
prefix / mask   {ip-address|type interface-path-id [{ip-address|type interface-path-id}] [track
track-object-name] [ tunnel-id tunnel-id] [vrflabel vrf-label] [distance] [description text] [tag tag]
[permanent]}
no prefix / mask {ip-address|type interface-path-id [{ip-address|type interface-path-id}] [track
track-object-name] [ tunnel-id tunnel-id] [vrflabel vrf-label] [distance] [description text] [tag tag]
[permanent]}
```

### Syntax Description

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>prefix / mask</td>
<td>IP route prefix and prefix mask for the destination.</td>
</tr>
<tr>
<td></td>
<td>The network mask can be specified in either of two ways:</td>
</tr>
<tr>
<td></td>
<td>• The network mask can be a four-part, dotted-decimal address. For example,</td>
</tr>
<tr>
<td></td>
<td>255.0.0.0 indicates that each bit equal to 1 means the corresponding address bit</td>
</tr>
<tr>
<td></td>
<td>is a network address.</td>
</tr>
<tr>
<td></td>
<td>• The network mask can be indicated as a slash (/) and number. For example,</td>
</tr>
<tr>
<td></td>
<td>/8 indicates that the first 8 bits of the mask are 1s, and the corresponding bits</td>
</tr>
<tr>
<td></td>
<td>of the address are the network address.</td>
</tr>
<tr>
<td>ip-address</td>
<td>IP address of the next hop that can be used to reach that network.</td>
</tr>
<tr>
<td></td>
<td>• For IPv4 address–the IP address is required, not optional, if the interface type</td>
</tr>
<tr>
<td></td>
<td>and interface-path-id arguments are not specified. You can specify an IP address</td>
</tr>
<tr>
<td></td>
<td>and an interface type and interface path.</td>
</tr>
<tr>
<td></td>
<td>• For IPv6 link-local address–the interface type and interface-path-id arguments</td>
</tr>
<tr>
<td></td>
<td>are required. The route is not valid, if the interface type and interface-path-id</td>
</tr>
<tr>
<td></td>
<td>arguments are not specified.</td>
</tr>
<tr>
<td>type</td>
<td>Interface type. For more information, use the question mark (?) online help function.</td>
</tr>
<tr>
<td>interface-path-id</td>
<td>Physical interface or virtual interface.</td>
</tr>
<tr>
<td></td>
<td>Use the <code>show interfaces</code> command to see a list of all interfaces currently configured on the router.</td>
</tr>
<tr>
<td></td>
<td>For more information about the syntax for the router, use the question mark ( ? ) online help function.</td>
</tr>
<tr>
<td></td>
<td>A forwarding router's IP address or an interface or virtual interface path ID can be configured, in any order.</td>
</tr>
<tr>
<td>distance</td>
<td>(Optional) Administrative distance. Range is 1 to 254.</td>
</tr>
<tr>
<td>description text</td>
<td>(Optional) Specifies a description of the static route.</td>
</tr>
</tbody>
</table>
**Static Routing Commands**

**route (static)**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tag</td>
<td>(Optional) Specifies a tag value that can be used as a match for controlling redistribution using route policies. Range is 1 to 4294967295.</td>
</tr>
<tr>
<td>permanent</td>
<td>(Optional) Specifies that the route is not removed from the routing table, even if the next-hop interface shuts down or next-hop IP address is not reachable.</td>
</tr>
<tr>
<td>track</td>
<td>Enables object tracking for static route.</td>
</tr>
<tr>
<td>track-object-name</td>
<td></td>
</tr>
<tr>
<td>tunnel-id</td>
<td>Specifies a Tunnel ID.</td>
</tr>
</tbody>
</table>

**Command Default**

No static route is established.

**Command Modes**

Address family configuration

**Command History**

Release 6.0  This command was introduced.

**Usage Guidelines**

A static route is appropriate when the software cannot dynamically build a route to the destination.

Static routes have a default administrative distance of 1, in which a low number indicates a preferred route. By default, static routes are preferred to routes learned by routing protocols. You can configure an administrative distance with a static route if you want the static route to be overridden by dynamic routes. For example, you could have routes installed by the Open Shortest Path First (OSPF) protocol with an administrative distance of 120. To have a static route that would be overridden by an OSPF dynamic route, specify an administrative distance greater than 120.

The routing table considers the static routes that point to an interface as “directly connected.” Directly connected networks are advertised by IGP routing protocols if a corresponding `interface` command is contained under the router configuration stanza of that protocol.

**Task ID**

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>static</td>
<td>read, write</td>
</tr>
</tbody>
</table>

**Examples**

The following example shows how to configure IPv6 unicast address family static routes:

```
RP/0/RP0/CPU0:router(config)# router static
RP/0/RP0/CPU0:router(config-static)# address-family ipv6 unicast
RP/0/RP0/CPU0:router(config-static-afi)# 2b11::327a::7b00/120 tenGigE 0/2/0/7
RP/0/RP0/CPU0:router(config-static-afi)# 2b11::327a::7b00/120 tenGigE 0/6/0/0
RP/0/RP0/CPU0:router(config-static-afi)# 2b11::327a::7b00/120 2b11::2f01:4c
RP/0/RP0/CPU0:router(config-static-afi)# 2b11::327a::7b00/120 2b11::2f01:4d
RP/0/RP0/CPU0:router(config-static-afi)# 2b11::327a::7b00/120 2b11::2f01:4e
RP/0/RP0/CPU0:router(config-static-afi)# 2b11::327a::7b00/120 2b11::2f01:50
```
router static

To enter static router configuration mode, use the `router static` command in XR Config mode. To remove all static route configurations and terminate the static routing process, use the `no` form of this command.

```
router static
no router static
```

Syntax Description

This command has no arguments or keywords.

Command Default

No static routing process is enabled.

Command Modes

XR Config mode

Command History

Release 6.0 This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

<table>
<thead>
<tr>
<th>Task ID</th>
<th>Operations</th>
</tr>
</thead>
<tbody>
<tr>
<td>static</td>
<td>read, write</td>
</tr>
<tr>
<td>bgp, ospf, isis, or network</td>
<td>read, write</td>
</tr>
</tbody>
</table>

Examples

The following example shows how to enter static router configuration mode:

```
RP/0/RP0/CPU0:router(config)# router static
RP/0/RP0/CPU0:router(config-static)#
```