# IP Addresses and Services Command Reference for Cisco NCS 6000 Series Routers

**First Published:** 2016-01-01

**Last Modified:** 2021-07-15

**Americas Headquarters**
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

# CONTENTS

# Preface

The *IP Addresses and Services Command Reference for Cisco NCS 6000 Series Routers*contains commands related to IP addresses and services features.

The preface contains the following sections:



## Changes to This Document

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

***Table 1: Changes to this Document***

| Date | Change Summary |
|---|---|
| July 2021 | Republished with documentation updates for Cisco IOS XR Release 7.4.1 features. |
| March 2018 | Republished with documentation updates for Cisco IOS XR Release 6.3.2 and 6.4.1 features. |
| September 2017 | Republished with documentation updates for Cisco IOS XR Release 6.3.1 features. |
| July 2017 | Republished with documentation updates for Cisco IOS XR Release 6.2.2 features. |
| November 2016 | Republished with documentation updates for Cisco IOS XR Release 6.1.2 features. |
| January 2015 | Initial release of the cumulative command reference document that covers all updates from Release 5.0.0 onwards. |

# 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.

# Access List Commands

This module describes the Cisco IOS XR software commands used to configure IP Version 4 (IPv4) and IP Version 6 (IPv6) access lists.

An access control list (ACL) consists of one or more access control entries (ACEs) that collectively define the network traffic profile. This profile can then be referenced by Cisco IOS XR Software software features such as traffic filtering, priority or custom queueing, and dynamic access control. Each ACL includes an action element (permit or deny) and a filter element based on criteria such as source address, destination address, protocol, and protocol-specific parameters.

For detailed information about ACL concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# clear access-list ipv4

To clear IPv4 access list counters, use the **clear access-list ipv4** command in XR EXEC mode .

**clear access-list ipv4***access-list name* `[ sequence-number | hardware { ingress | egress} interface type interface-path-id [sequence number] location node-id | location node-id | sequence number location node-id]`

## Syntax Description

| | |
|---|---|
| access-list-name | Name of a particular IPv4 access list. The name cannot contain a spaces or quotation marks, but can include numbers. |
| sequence-number | (Optional) Specific sequence number with which counters are cleared for an access list. Range is 1 to 2147483646. |
| hardware | Identifies the access list as an access group for an interface. |
| ingress | Specifies an inbound direction. |
| egress | Specifies an outbound direction. |
| interface | (Optional) Clears the interface statistics. |
| type | Interface type. For more information, use the question mark (?) online help function. |
| interface-path-id | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Clears hardware resource counters from the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **sequence** *number* | (Optional) Clears counters for an access list with a specific sequence number. Range is 1 to 2147483646. |

## Command Default

The default clears the specified IPv4 access list.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **clear access-list ipv4** command to clear counters for a specified configured access list. Use a sequence number to clear counters for an access list with a specific sequence number.

Use the **hardware** keyword to clear counters for an access list that was enabled using the **ipv4 access-group** command.

Use an asterisk ( ***)** in place of the *access-list-name* argument to clear all access lists.

**Note** An access list can be shared among multiple interfaces. Clearing hardware counters clears all counters for all interfaces that use the specified access list in a given direction (ingress or egress).

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| acl | read, write |
| bgp | read, write, execute |

## Examples

In the following example, counters for an access list named *marketing* are cleared:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 marketing

ipv4 access-list marketing
  10 permit ip 192.168.34.0 0.0.0.255 any (51 matches)
  20 permit ip 172.16.0.0 0.0.255.255 any (26 matches)
  30 deny tcp host 172.16.0.0 eq bgp host 192.168.202.203 30 (5 matches)

RP/0/RP0/CPU0:router# clear access-list ipv4 marketing

RP/0/RP0/CPU0:router# show access-lists ipv4 marketing

ipv4 access-list marketing
  10 permit ip 192.168.34.0 0.0.0.255 any
  20 permit ip 172.16.0.0 0.0.255.255 any
  30 deny tcp host 172.16.0.0 eq bgp host 192.168.202.203 30
```

In the following example, counters for an access list named acl_hw_1 in the outbound direction are cleared:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 acl_hw_1 hardware egress location 0/2/cp0

ipv4 access-list acl_hw_1
  10 permit icmp 192.168.36.0 0.0.0.255 any (251 hw matches)
  20 permit ip 172.16.3.0 0.0.255.255 any (29 hw matches)
  30 deny tcp any any (58 hw matches)

RP/0/RP0/CPU0:router# clear access-list ipv4 acl_hw_1 hardware egress location 0/2/cp0

RP/0/RP0/CPU0:router# show access-lists ipv4 acl_hw_1 hardware egress location 0/2/cp0

ipv4 access-list acl_hw_1
  10 permit icmp 192.168.36.0 0.0.0.255 any
  20 permit ip 172.16.3.0 0.0.255.255 any
  30 deny tcp any any
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 access-group, on page 27 | Filters incoming or outgoing IPv4 traffic on an interface. |

| Command | Description |
| --- | --- |
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| resequence access-list ipv4 , on page 58 | Renumbers an existing statement and increments subsequent statements to allow a new IPv4 access list statements. |

# clear access-list ipv6

To clear IPv6 access list counters, use the **clear access-list ipv6** command in XR EXEC mode.

**clear access-list ipv4***access-list name* `[ sequence-number | hardware { ingress | egress} interface type interface-path-id [sequence number] location node-id | location node-id | sequence number location node-id]`

## Syntax Description

| | |
|---|---|
| access-list-name | Name of a particular IPv6 access list. The name cannot contain a spaces or quotation marks, but can include numbers. |
| sequence-number | (Optional) Specific sequence number for a particular access control entry (ACE) with which counters are cleared for an access list. Range is 1 to 2147483644. |
| hardware | (Optional) Identifies the access list as an access group for an interface. |
| ingress | (Optional) Specifies an inbound direction. |
| egress | (Optional) Specifies an outbound direction. |
| interface | (Optional) Clears the interface statistics. |
| type | Optional) Interface type. For more information, use the question mark (?) online help function. |
| *instance* | Physical interface or virtual interface. |
| *interface-path-id* | **Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br><br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Clears counters for an access list enabled on a card interface. The *node-id* argument is entered in the rack/slot/module notation. |
| **sequence** *number* | (Optional) Specifies a specific sequence number that clears access list counters. Range is 1 to 2147483644. |

## Command Default

The default clears the specified IPv6 access list.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **clear access-list ipv6** command is similar to the **clear access-list ipv4** command, except that it is IPv6-specific.

Use the **clear access-list ipv6** command to clear counters for a specified configured access list. Use a sequence number to clear counters for an access list with a specific sequence number

Use the**hardware** keyword to clear counters for an access list that was enabled using the **ipv6 access-group** command.

Use an asterisk (*****) in place of the *access-list-name* argument to clear all access lists.

**Note** An access list can be shared among multiple interfaces. Clearing hardware counters clears all counters for all interfaces that use the specified access list in a given direction (ingress or egress).

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| acl | read, write |
| network | read, write |

## Examples

In the following example, counters for an access list named *marketing* are cleared:

```
RP/0/RP0/CPU0:router# show access-lists ipv6 marketing
ipv6 access-list marketing
  10 permit ipv6 3333:1:2:3::/64 any (51 matches)
  20 permit ipv6 4444:1:2:3::/64 any (26 matches)
  30 permit ipv6 5555:1:2:3::/64 any (5 matches)
RP/0/RP0/CPU0:router# clear access-list ipv6 marketing
RP/0/RP0/CPU0:router# show access-lists ipv6 marketing
ipv6 access-list marketing
  10 permit ipv6 3333:1:2:3::/64 any
  20 permit ipv6 4444:1:2:3::/64 any
  30 permit ipv6 5555:1:2:3::/64 any
```

In the following example, counters for an access list named acl_hw_1 in the outbound direction are cleared:

```
RP/0/RP0/CPU0:router# show access-lists ipv6 acl_hw_1 hardware egress location 0/2/cp0
ipv6 access-list acl_hw_1
  10 permit ipv6 3333:1:2:3::/64 any (251 hw matches)
  20 permit ipv6 4444:1:2:3::/64 any (29 hw matches)
  30 deny tcp any any (58 hw matches)
RP/0/RP0/CPU0:router# clear access-list ipv6 acl_hw_1 hardware egress location 0/2/cp0
RP/0/RP0/CPU0:router# show access-lists ipv6 acl_hw_1 hardware egress location 0/2/cp0
ipv6 access-list acl_hw_1
  10 permit ipv6 3333:1:2:3::/64 any
  20 permit ipv6 4444:1:2:3::/64 any
  30 deny tcp any any
```

**Related Commands**

| Command | Description |
|---|---|
| ipv6 access-list, on page 34 | Defines an IPv6 access list and enters IPv6 access list configuration mode. |

# copy access-list ipv4

To create a copy of an existing IPv4 access list, use the **copy access-list ipv4** command in XR EXEC mode.

**copy access-list ipv4** *source-acl destination-acl*

## Syntax Description

| | |
|---|---|
| source-acl | Name of the access list to be copied. |
| destination-acl | Name of the destination access list where the contents of the *source-acl* argument is copied. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **copy access-list ipv4** command to copy a configured access list. Use the *source-acl* argument to specify the access list to be copied and the *destination-acl* argument to specify where to copy the contents of the source access list. The *destination-acl* argument must be a unique name; if the *destination-acl* argument name exists for an access list or prefix list, the access list is not copied. The **copy access-list ipv4** command checks that the source access list exists then checks the existing list names to prevent overwriting existing access lists or prefix lists.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| filesystem | execute |

## Examples

In the following example, a copy of access list list-1 is created:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 list-1

ipv4 access-list list-1
  10 permit tcp any any log
  20 permit ip any any
RP/0/RP0/CPU0:router# copy access-list ipv4 list-1 list-2
RP/0/RP0/CPU0:router# show access-lists ipv4 list-2
ipv4 access-list list-2
  10 permit tcp any any log
  20 permit ip any any
```

In the following example, copying the access list list-1 to list-3 is denied because a list-3 access list already exists:

```
RP/0/RP0/CPU0:router# copy access-list ipv4 list-1 list-3

list-3 exists in access-list

RP/0/RP0/CPU0:router# show access-lists ipv4 list-3

ipv4 access-list list-3
  10 permit ip any any
  20 deny tcp any any log
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| show access-lists ipv4 , on page 63 | Displays the contents of all current IPv4 access lists. |

# copy access-list ipv6

To create a copy of an existing IPv6 access list, use the **copy access-list ipv6** command in XR EXEC mode .

**copy access-list ipv6** *source-acl destination-acl*

**Syntax Description**

| | |
|---|---|
| source-acl | Name of the access list to be copied. |
| destination-acl | Destination access list where the contents of the *source-acl* argument is copied. |

**Command Default**

No default behavior or value

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

Use the **copy access-list ipv6** command to copy a configured access list. Use the *source-acl* argument to specify the access list to be copied and the *destination-acl* argument to specify where to copy the contents of the source access list. The *destination-acl* argument must be a unique name; if the *destination-acl* argument name exists for an access list or prefix list, the access list is not copied. The **copy access-list ipv6** command checks that the source access list exists then checks the existing list names to prevent overwriting existing access lists or prefix lists.

**Task ID**

| Task ID | Operations |
|---|---|
| acl | read, write |
| filesystem | execute |

**Examples**

In this example, a copy of access list list-1 is created:

```
RP/0/RP0/CPU0:router# show access-lists ipv6 list-1

ipv6 access-list list-1
  10 permit tcp any any log
  20 permit ipv6 any any

RP/0/RP0/CPU0:router# copy access-list ipv6 list-1 list-2

RP/0/RP0/CPU0:router# show access-lists ipv6 list-2

ipv6 access-list list-2
  10 permit tcp any any log
  20 permit ipv6 any any
```

In this example, copying access list list-1 to list-3 is denied because a list-3 access list already exists:

```
RP/0/RP0/CPU0:router# copy access-list ipv6 list-1 list-3

list-3 exists in access-list

RP/0/RP0/CPU0:router# show access-lists ipv6 list-3
ipv6 access-list list-3
  10 permit ipv6 any any
  20 deny tcp any any log
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 access-list, on page 34 | Defines an IPv6 access list and enters IPv6 access list configuration mode. |
| show access-lists ipv6, on page 70 | Displays the contents of all current IPv6 access lists. |

# deny (IPv4)

To set conditions for an IPv4 access list, use the **deny** command in access list configuration mode. There are two versions of the **deny** command: **deny** (source), and **deny** (protocol). To remove a condition from an access list, use the **no** form of this command.

[*sequence-number*] **deny** *source* [*source-wildcard*] **counter** *counter-name* [{**log** | **log-input**}]
*[sequence-number]***deny***protocol source source-wildcard destination destination-wildcard* [**precedence***precedence*] [**dscp***dscp*] [**fragments**] [ *packet-length operator packet-length value*] [ **log | log-input**] [**ttl** *ttl value [value1....value2]*]
**no** *sequence-number*

**Internet Control Message Protocol (ICMP)**
[*sequence-number*] **deny icmp** *source source-wildcard destination destination-wildcard* [*icmp-type*] [*icmp-code*] [**precedence** *precedence*] [**dscp** *dscp*] [**fragments**] [{**log** | **log-input**}][**icmp-off**]

**Internet Group Management Protocol (IGMP)**
[*sequence-number*] **deny igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**dscp** *value*] [**fragments**] [{**log** | **log-input**}]

**User Datagram Protocol (UDP)**
[*sequence-number*] **deny udp** *source source-wildcard* [*operator* {*portprotocol-port*}] *destination destination-wildcard* [*operator* {*portprotocol-port*}] [**precedence** *precedence*] [**dscp** *dscp*] [**fragments**] [{**log** | **log-input**}]

## Syntax Description

| | |
|---|---|
| sequence-number | (Optional) Number of the **deny** statement in the access list. This number determines the order of the statements in the access list. The number can be from 1 to 2147483646. (By default, the first statement is number 10, and the subsequent statements are incremented by 10.) Use the **resequence access-list** command to change the number of the first statement and increment subsequent statements of a configured access list. |
| source | Number of the network or host from which the packet is being sent. There are three alternative ways to specify the source:<br>• Use a 32-bit quantity in four-part dotted-decimal format.<br>• Use the **any** keyword as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255.<br>• Use the **host** *source* combination as an abbreviation for a *source* and *source-wildcard* of *source* 0.0.0.0. |
| source-wildcard | Wildcard bits to be applied to the source. There are three alternative ways to specify the source wildcard:<br>• Use a 32-bit quantity in four-part dotted-decimal format. Place ones in the bit positions you want to ignore.<br>• Use the **any** keyword as an abbreviation for a *source* and *source-wildcard* of 0.0.0.0 255.255.255.255.<br>• Use the **host** *source* combination as an abbreviation for a *source* and *source-wildcard* of *source* 0.0.0.0. |

| | |
|---|---|
| protocol | Name or number of an IP protocol. It can be one of the keywords **ahp** , **esp** , **eigrp** , **gre** , **icmp** , **igmp** , **igrp** , **ip** , **ipinip** , **nos** , **ospf** , **pim** , **pcp** , , **tcp** , or **udp** , or an integer from 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the **ip** keyword. ICMP, SCTP, and TCP allow further qualifiers, which are described later in this table. |
| destination | Number of the network or host to which the packet is being sent. There are three alternative ways to specify the destination:<br>• Use a 32-bit quantity in four-part dotted-decimal format.<br>• Use the **any** keyword as an abbreviation for the *destination* and *destination-wildcard* of 0.0.0.0 255.255.255.255.<br>• Use the **host** *destination* combination as an abbreviation for a *destination* and *destination-wildcard* of *destination* 0.0.0.0. |
| destination-wildcard | Wildcard bits to be applied to the destination. There are three alternative ways to specify the destination wildcard:<br>• Use a 32-bit quantity in four-part dotted-decimal format. Place ones in the bit positions you want to ignore.<br>• Use the **any** keyword as an abbreviation for a *destination* and *destination-wildcard* of 0.0.0.0 255.255.255.255.<br>• Use the **host** *destination* combination as an abbreviation for a *destination* and *destination-wildcard* of *destination* 0.0.0.0. |
| **precedence** *precedence* | (Optional) Packets can be filtered by precedence level (as specified by a number from 0 to 7) or by the following names:<br>• **routine** –Match packets with routine precedence (0)<br>• **priority** –Match packets with priority precedence (1)<br>• **immediate** –Match packets with immediate precedence (2)<br>• **flash** –Match packets with flash precedence (3)<br>• **flash-override** –Match packets with flash override precedence (4)<br>• **critical** –Match packets with critical precedence (5)<br>• **internet** –Match packets with internetwork control precedence (6)<br>• **network** –Match packets with network control precedence (7) |

| | |
|---|---|
| **dscp** *dscp* | (Optional) Differentiated services code point (DSCP) provides quality of service control. The values for *dscp* are as follows:<br>• 0—63–Differentiated services codepoint value<br>• af11–Match packets with AF11 dscp (001010)<br>• af12–Match packets with AF12 dscp (001100)<br>• af13–Match packets with AF13 dscp (001110)<br>• af21–Match packets with AF21 dscp (010010)<br>• af22–Match packets with AF22 dscp (010100)<br>• af23–Match packets with AF23 dscp (010110)<br>• af31–Match packets with AF31 dscp (011010)<br>• af32–Match packets with AF32 dscp (011100)<br>• af33–Match packets with AF33 dscp (011110)<br>• af41–Match packets with AF41 dscp (100010)<br>• af42—Match packets with AF42 dscp (100100)<br>• af43–Match packets with AF43 dscp (100110)<br>• cs1–Match packets with CS1(precedence 1) dscp (001000)<br>• cs2–Match packets with CS2(precedence 2) dscp (010000)<br>• cs3–Match packets with CS3(precedence 3) dscp (011000)<br>• cs4–Match packets with CS4(precedence 4) dscp (100000)<br>• cs5–Match packets with CS5(precedence 5) dscp (101000)<br>• cs6–Match packets with CS6(precedence 6) dscp (110000)<br>• cs7–Match packets with CS7(precedence 7) dscp (111000)<br>• default–Default DSCP (000000)<br>• ef–Match packets with EF dscp (101110) |
| fragments | (Optional) Causes the software to examine fragments of IPv4 packets when applying this access list entry. When this keyword is specified, fragments are subject to the access list entry. |
| packet-length operator | (Optional) Packet length operator used for filtering. |
| packet-length value | (Optional) Packet length used to match only packets in the range of the length. |
| log | (Optional) Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.)<br>The message includes the access list number, whether the packet was permitted or denied; the protocol, whether it was TCP, UDP, ICMP, or a number; and, if appropriate, the source and destination addresses and source and destination port numbers. The message is generated for the first packet that matches a flow, and then at 5-minute intervals, including the number of packets permitted or denied in the prior 5-minute interval. |
| log-input | (Optional) Provides the same function as the **log** keyword, except that the logging message also includes the input interface. |
| ttl | (Optional) Turns on matching against time-to-life (TTL) value. |

| | |
|---|---|
| ttl value1 value2 | (Optional) TTL value used for filtering. Range is 1 to 255.<br>If only *value1* is specified, the match is against this value.<br>If both *value1* and *value2* are specified, the packet TTL is matched against the range of TTLs between *value1* and *value2* . |
| icmp-off | (Optional) Turns off ICMP generation for denied packets. |
| icmp-type | (Optional) ICMP message type for filtering ICMP packets. Range is from 0 to 255. |
| icmp-code | (Optional) ICMP message code for filtering ICMP packets. Range is from 0 to 255. |
| igmp-type | (Optional) IGMP message type (0 to 15) or message name for filtering IGMP packets, as follows:<br>• dvmrp<br>• host-query<br>• host-report<br>• mtrace<br>• mtrace-response<br>• pim<br>• precedence<br>• trace<br>• v2-leave<br>• v2-report<br>• v3-report |
| operator | (Optional) Operator is used to compare source or destination ports. Possible operands are **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range).<br>If the operator is positioned after the *source* and *source-wildcard* values, it must match the source port.<br>If the operator is positioned after the *destination* and *destination-wildcard* values, it must match the destination port.<br>If the operator is positioned after the **ttl** keyword, it matches the TTL value.<br>The **range** operator requires two port numbers. All other operators require one port number. |
| protocol-port | Name of a TCP or UDP port. TCP and UDP port names are listed in the "Usage Guidelines" section.<br>TCP port names can be used only when filtering TCP. UDP port names can be used only when filtering UDP. |
| established | (Optional) For the TCP protocol only: Indicates an established connection. |
| match-any | (Optional) For the TCP protocol only: Filters on any combination of TCP flags. |
| match-all | (Optional) For the TCP protocol only: Filters on all TCP flags. |

| | |
|---|---|
| + \| - | (Required) For the TCP protocol **match-any** , **match-all** : Prefix *flag-name* with + or - . Use the + *flag-name* argument to match packets with the TCP flag set. Use the - *flag-name* argument to match packets when the TCP flag is not set. |
| flag-name | (Required) For the TCP protocol **match-any** , **match-all** . Flag names are: ack, fin, psh, rst, syn. |

## Command Default

There is no specific condition under which a packet is denied passing the IPv4 access list.

ICMP message generation is enabled by default.

## Command Modes

IPv4 access list configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **deny** command following the **ipv4 access-list** command to specify conditions under which a packet cannot pass the access list.

By default, the first statement in an access list is number 10, and the subsequent statements are incremented by 10.

You can add **permit**, **deny**, or **remark** statements to an existing access list without retyping the entire list. To add a new **s**tatement anywhere other than at the end of the list, create a new statement with an appropriate entry number that falls between two existing entry numbers to indicate where it belongs.

If you want to add a statement between two consecutively numbered statements (for example, between lines 10 and 11), first use the **resequence access-list** command to renumber the first statement and increment the entry number of each subsequent statement. The *increment* argument causes new, unused line numbers between statements. Then add a new statement with the *entry-number* argument, specifying where it belongs in the access list.

The following is a list of precedence names:

- critical
- flash
- flash-override
- immediate
- internet
- network
- priority
- routine

The following is a list of ICMP message type names:

- administratively-prohibited
- alternate-address
- conversion-error
- dod-host-prohibited
- dod-net-prohibited
- echo

- echo-reply
- general-parameter-problem
- host-isolated
- host-precedence-unreachable
- host-redirect
- host-tos-redirect
- host-tos-unreachable
- host-unknown
- host-unreachable
- information-reply
- information-request
- mask-reply
- mask-request
- mobile-redirect
- net-redirect
- net-tos-redirect
- net-tos-unreachable
- net-unreachable
- network-unknown
- no-room-for-option
- option-missing
- packet-too-big
- parameter-problem
- port-unreachable
- precedence-unreachable
- protocol-unreachable
- reassembly-timeout
- redirect
- router-advertisement
- router-solicitation
- source-quench
- source-route-failed
- time-exceeded
- timestamp-reply
- timestamp-request
- traceroute
- ttl-exceeded
- unreachable

The following is a list of TCP port names that can be used instead of port numbers. Refer to the current *Assigned Numbers* RFC to find a reference to these protocols. You can find port numbers corresponding to these protocols by typing a ? in the place of a port number.

- bgp
- chargen
- cmd
- daytime

- discard
- domain
- echo
- exec
- finger
- ftp
- ftp-data
- gopher
- hostname
- ident
- irc
- klogin
- kshell
- login
- lpd
- nntp
- pim-auto-rp
- pop2
- pop3
- smtp
- sunrpc
- tacacs
- talk
- telnet
- time
- uucp
- whois
- www

The following UDP port names can be used instead of port numbers. Refer to the current *Assigned Numbers* RFC to find a reference to these protocols. You can find port numbers corresponding to these protocols by typing a ? in the place of a port number.

- biff
- bootpc
- bootps
- discard
- dnsix
- domain
- echo
- isakmp
- mobile-ip
- nameserver
- netbios-dgm
- netbios-ns
- netbios-ss
- ntp

- pim-auto-rp
- rip
- snmp
- snmptrap
- sunrpc
- syslog
- tacacs
- talk
- tftp
- time
- who
- xdmcp

Use the following flags in conjunction with the **match-any** and **match-all** keywords and the + and - signs to select the flags to display:

- ack
- fin
- psh
- rst
- syn

For example, **match-all** + *ack* + *syn* displays TCP packets with both the ack *and* syn flags set, or **match-any** + *ack* - *syn* displays the TCP packets with the ack set *or* the syn not set.

**Note** If any ACE in an ACL contains ABF clause, this ACL cannot be applied at any non-zero compression level.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| acl | read, write |

## Examples

This example shows how to set a deny condition for an access list named Internet filter:

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list Internetfilter
RP/0/RP0/CPU0:router(config-ipv4-acl)# 10 deny 192.168.34.0 0.0.0.255
RP/0/RP0/CPU0:router(config-ipv4-acl)# 20 deny 172.16.0.0 0.0.255.255
RP/0/RP0/CPU0:router(config-ipv4-acl)# 25 deny tcp host 172.16.0.0 eq bgp host 192.168.202.203
 range 1300 1400
RP/0/RP0/CPU0:router(config-ipv4-acl)# permit 10.0.0.0 0.255.255.255
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 access-group, on page 27 | Filters incoming or outgoing IPv4 traffic on an interface. |
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| permit (IPv4) , on page 38 | Sets the permit conditions for an IPv4 access list |
| remark (IPv4) , on page 54 | Inserts a helpful remark about an IPv4 access list entry. |
| resequence access-list ipv4 , on page 58 | Changes the starting entry number of the first statement in an existing IPv4 access list, and the number by which subsequent statements are incremented. |
| show access-lists ipv4 , on page 63 | Displays the contents of all current IPv4 access lists. |

# deny (IPv6)

To set deny conditions for an IPv6 access list, use the **deny** command in IPv6 access list configuration mode. To remove the deny conditions, use the **no** form of this command.

*[sequence-number]* **deny** *protocol* {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* {*port* | *protocol-port*}] [**dscp***value*] [**routing**] [**authen**] [**destopts**] [ **fragments**] [*packet-length operator packet-length value* ] [ **log | log-input**] [**ttl** *operator ttl value* ]
**no** *sequence-number*

**Internet Control Message Protocol (ICMP)**
*[ sequence-number]***deny icmp** [*icmp-type*] [ *icmp-code*] [**dscp** *value*] [ routing] [**authen**] [**destopts**] [ **fragments**] [ **log**] [**log-input**] [**icmp-off**]

**Transmission Control Protocol (TCP)**
*[sequence-number]***deny tcp** [*operator*{*port* | *protocol-port*}] [*operator*{*port* | *protocol* | *port*}] [**dscp***value*] [**routing**] [**authen**] [**destopts**] [**fragments**] [**established**] {**match-any | match-all | + | -**} [*flag-name*] [**log**] [**log-input**]

**User Datagram Protocol (UDP)**
*[sequence-number]***deny tcp** [*operator*{*port* | *protocol-port*}] [*operator*{*port* | *protocol* | *port*}] [**dscp***value*] [**routing**] [**authen**] [**destopts**] [**fragments**] [**established**] [*flag-name*] [**log**] [**log-input**]

## Syntax Description

| | |
|---|---|
| sequence-number | (Optional) Number of the **deny** statement in the access list. This number determines the order of the statements in the access list. Range is 1 to 2147483646. (By default, the first statement is number 10, and the subsequent statements are incremented by 10.) Use the **resequence access-list** command to change the number of the first statement and increment subsequent statements of a configured access list. |
| protocol | Name or number of an Internet protocol. It can be one of the keywords **ahp** , **esp** , **icmp** , **ipv6** , **pcp** , **sctp** , **tcp** , or **udp** , or an integer in the range from 0 to 255 representing an IPv6 protocol number. |
| *source-ipv6-prefix / prefix-length* | The source IPv6 network or class of networks about which to set deny conditions.<br>This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **any** | An abbreviation for the IPv6 prefix ::/0. |
| **host** *destination-ipv6-address* | Destination IPv6 host address about which to set deny conditions.<br>This *destination-ipv6-address* argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |

| | |
|---|---|
| *operator* {*port* \| *protocol-port*} | (Optional) Operand that compares the source or destination ports of the specified protocol. Operands are **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range).<br>If the operator is positioned after the *source-ipv6-prefix / prefix-length* argument, it must match the source port.<br>If the operator is positioned after the *destination-ipv6-prefix / prefix-length* argument, it must match the destination port.<br>The **range** operator requires two port numbers. All other operators require one port number.<br>The *port* argument is the decimal number of a TCP or UDP port. Range is 0 to 65535. The *protocol-port* argument is the name of a TCP or UDP port. TCP port names can be used only when filtering TCP. UDP port names can be used only when filtering UDP. |
| *destination-ipv6-prefix / prefix-length* | Destination IPv6 network or class of networks about which to set deny conditions.<br>This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **host** *destination-ipv6-address* | Destination IPv6 host address about which to set deny conditions.<br>This *destination-ipv6-address* argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **dscp** *value* | (Optional) Matches a differentiated services code point DSCP value against the traffic class value in the Traffic Class field of each IPv6 packet header. Range is 0 to 63. |
| routing | (Optional) Matches source-routed packets against the routing extension header within each IPv6 packet header. |
| authen | (Optional) Matches if the IPv6 authentication header is present. |
| destopts | (Optional) Matches if the IPv6 destination options header is present. |
| fragments | (Optional) Matches non-initial fragmented packets where the fragment extension header contains a nonzero fragment offset. The **fragments** keyword is an option only if the *operator* [ *port-number* ] arguments are not specified. |
| packet-length operator | (Optional) Packet length operator used for filtering. |
| packet-length value | (Optional) Packet length used to match only packets in the range of the length. |

| | |
|---|---|
| log | (Optional) Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.)<br>The message includes the access list name and sequence number, whether the packet was denied; the protocol, whether it was TCP, UDP, ICMP, or a number; and, if appropriate, the source and destination addresses and source and destination port numbers. The message is generated for the first packet that matches, and then at 5-minute intervals, including the number of packets denied in the prior 5-minute interval. |
| log-input | (Optional) Provides the same function as the **log** keyword, except that the logging message also includes the input interface. |
| ttl | (Optional) Turns on matching against time-to-life (TTL) value. |
| operator | (Optional) Operand that compares the source or destination ports of the specified protocol. Operands are **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range). |
| ttl value1 value2 | (Optional) TTL value used for filtering. Range is 1 to 255.<br>If only *value1* is specified, the match is against this value.<br>If both *value1* and *value2* are specified, the packet TTL is matched against the range of TTLs between *value1* and *value2* . |
| icmp-off | (Optional) Turns off ICMP generation for denied packets |
| icmp-type | (Optional) ICMP message type for filtering ICMP packets. ICMP packets can be filtered by ICMP message type. Range is 0 to 255. |
| icmp-code | (Optional) ICMP message code for filtering ICMP packets. ICMP packets that are filtered by ICMP message type can also be filtered by the ICMP message code. Range is 0 to 255. |
| established | (Optional) For the TCP protocol only: Indicates an established connection. |
| match-any | (Optional) For the TCP protocol only: Filters on any combination of TCP flags. |
| match-all | (Optional) For the TCP protocol only: Filters on all TCP flags. |
| + \| - | (Required) For the TCP protocol **match-any** , **match-all** : Prefix *flag-name* with + or - . Use the + *flag-name* argument to match packets with the TCP flag set. Use the - *flag-name* argument to match packets when the TCP flag is not set. |
| flag-name | (Required) For the TCP protocol **match-any** , **match-all** . Flag names are: ack, fin, psh, rst, syn. |

**Command Default**

No IPv6 access list is defined.

ICMP message generation is enabled by default.

**Command Modes**

IPv6 access list configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **deny** (IPv6) command is similar to the **deny** (IPv4) command, except that it is IPv6-specific.

Use the **deny** (IPv6) command following the **ipv6 access-list** command to define the conditions under which a packet passes the access list.

**Note** If any ACE in an ACL contains ABF clause, this ACL cannot be applied at any non-zero compression level.

Specifying **ipv6** for the *protocol* argument matches against the IPv6 header of the packet.

By default, the first statement in an access list is number 10, and the subsequent statements are numbered in increments of 10.

You can add permit, deny, or remark statements to an existing access list without retyping the entire list. To add a new statement anywhere other than at the end of the list, create a new statement with an appropriate entry number that falls between two existing entry numbers to indicate where it belongs.

Both the *source-ipv6-prefix/prefix-length* and *destination-ipv6-prefix/prefix-length* arguments are used for traffic filtering (the source prefix filters traffic based upon the traffic source; the destination prefix filters traffic based upon the traffic destination).

**Note** IPv6 prefix lists, not access lists, should be used for filtering routing protocol prefixes.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

The following example shows how to configure the IPv6 access list named toCISCO and applies the access list to outbound traffic on Packet-over-SONET (POS) interface 0/2/0/2. Specifically, the first deny entry in the list keeps all packets that have a destination TCP port number greater than 5000 from exiting out of POS interface 0/2/0/2. The second deny entry in the list keeps all packets that have a source UDPo port number less than 5000 from exiting out of POS interface 0/2/0/2. The second deny entry also logs all matches to the console. The first permit entry in the list permits all ICMP packets to exit out of POS interface 0/2/0/2. The second permit entry in the list permits all other traffic to exit out of POS interface 0/2/0/2. The second permit entry is necessary because an implicit deny all condition is at the end of each IPv6 access list.

```
RP/0/RP0/CPU0:router(config)# ipv6 access-list toCISCO
RP/0/RP0/CPU0:router(config-ipv6-acl)# deny tcp any any gt 5000
RP/0/RP0/CPU0:router(config-ipv6-acl)# deny ipv6 any lt 5000 any log
RP/0/RP0/CPU0:router(config-ipv6-acl)# permit icmp any any
```

```
RP/0/RP0/CPU0:router(config-ipv6-acl)# permit any any
RP/0/RP0/CPU0:router(config)# interface POS 0/2/0/2
RP/0/RP0/CPU0:router(config-if)# ipv6 access-group tOCISCO out
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 access-list, on page 34 | Defines an IPv6 access list and enters IPv6 access list configuration mode. |
| permit (IPv6) , on page 49 | Sets permit conditions for an IPv6 access list. |
| remark (IPv6) , on page 56 | Inserts a helpful remark about an IPv6 access list entry. |
| resequence access-list ipv6 , on page 60 | Changes the starting entry number of the first statement in an existing IPv6 access list, and the number by which subsequent statements are incremented. |

# ipv4 access-group

To control access to an interface, use the **ipv4 access-group** command in interface configuration mode. To remove the specified access group, use the **no** form of this command.

**ipv4 access-group** *access-list-name* {**ingress** | **egress**} [**hardware-count**] [**interface-statistics**]
**no ipv4 access-group** *access-list-name* {**ingress** | **egress**} [**hardware-count**] [**interface-statistics**]

## Syntax Description

| | |
|---|---|
| access-list-name | Name of an IPv4 access list as specified by an **ipv4 access-list** command. |
| ingress | Filters on inbound packets. |
| egress | Filters on outbound packets. |
| hardware-count | (Optional) Specifies to access a group's hardware counters. |
| interface-statistics | (Optional) Specifies per-interface statistics in the hardware. |

## Command Default

The interface does not have an IPv4 access list applied to it.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **ipv4 access-group** command to control access to an interface. To remove the specified access group, use the **no** form of the command. Use the *access-list-name* argument to specify a particular IPv4 access list. Use the **ingress** keyword to filter on inbound packets or the**egress** keyword to filter on outbound packets. Use the *hardware-count* argument to enable hardware counters for the access group.

Permitted packets are counted only when hardware counters are enabled using the *hardware-count* argument. Denied packets are counted whether hardware counters are enabled, or not.

Filtering of MPLS packets through common ACL and interface ACL is not supported.

**Note** For packet filtering applications using the **ipv4 access-group** command, packet counters are maintained in hardware for each direction. If an access group is used on multiple interfaces in the same direction, then packets are counted for each interface that has the *hardware-count* argument enabled.

If the access list permits the addresses, the software continues to process the packet. If the access list denies the address, the software discards the packet and returns an Internet Control Message Protocol (ICMP) host unreachable message.

If the specified access list does not exist, all packets are passed.

By default, the unique or per-interface ACL statistics are disabled.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| network | read, write |

## Examples

The following example shows how to apply filters on packets inbound and outbound from HundredGigE interface 0/7/0/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/7/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 access-group p-ingress-filter ingress
RP/0/RP0/CPU0:router(config-if)# ipv4 access-group p-egress-filter egress
```

The following example shows how to apply per-interface statistics in the hardware:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/2/0/2
RP/0/RP0/CPU0:router(config-if)# ipv4 access-group p-ingress-filter ingress
interface-statistics
```

## Related Commands

| Command | Description |
|---|---|
| clear access-list ipv4, on page 3 | Resets the IPv4 access list match counters. |
| deny (IPv4) , on page 13 | Sets the deny conditions for an ACE of an IPv4 access list. |
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| permit (IPv4) , on page 38 | Sets the permit conditions for an ACE of an IPv4 access list. |
| show access-lists ipv4 , on page 63 | Displays the contents of all current IPv4 access lists. |
| show ipv4 interface | Displays the usability status of interfaces configured for IPv4. |

# ipv4 access-list

To define an IPv4 access list by name, use the **ipv4 access-list** command in XR Config mode. To remove all entries in an IPv4 access list, use the **no** form of this command.

**ipv4 access-list** *name*

## Syntax Description

| | |
|---|---|
| *name* | Name of the access list. Names cannot contain a space or quotation marks. |

## Command Default

No IPv4 access list is defined.

## Command Modes

XR Config mode

## Usage Guidelines

Use the **ipv4 access-list** command to configure an IPv4 access list. This command places the router in access list configuration mode, in which the denied or permitted access conditions must be defined with the **deny** or **permit** command.

Use the **resequence access-list ipv4** command if you want to add a **permit** , **deny** , or **remark** statement between consecutive entries in an existing IPv4 access list. Specify the first entry number (the *base* ) and the increment by which to separate the entry numbers of the statements. The software renumbers the existing statements, thereby making room to add new statements with the unused entry numbers.

Use the **ipv4 access-group** command to apply the access list to an interface.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

The following example shows how to define a standard access list named Internetfilter:

```
Router(config)# ipv4 access-list Internetfilter
Router(config-if)# 10 permit 192.168.34.0 0.0.0.255
Router(config-if)# 20 permit 172.16.0.0 0.0.255.255
Router(config-if)# 30 permit 10.0.0.0 0.255.255.255
Router(config-if)# 39 remark Block BGP traffic from 172.16 net.
Router(config-if)# 40 deny tcp host 172.16.0.0 eq bgp host 192.168.202.203 range 1300 1400
```

## Related Commands

| Command | Description |
|---|---|
| show access-lists ipv4 | Displays the contents of all current IPv4 access lists. |

# ipv4 access-list log-update rate

To specify the rate at which IPv4 access lists are logged, use the **ipv4 access-list log-update rate** command in XR Config mode. To return the update rate to the default setting, use the **no** form of this command.

**ipv4 access-list log-update rate** *rate-number*
**no ipv4 access-list log-update rate** *rate-number*

## Syntax Description

| | |
|---|---|
| rate-number | Rate at which IPv4 access hit logs are generated per second on the router. Range is 1 to 1000. |

## Command Default

Default is 1.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The *rate-number* argument applies to all the IPv4 access-lists configured on the interfaces. That is, at any given time there can be between 1 and 1000 log entries for the system.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| acl | read, write |

## Examples

The following example shows how to configure a IPv4 access hit logging rate for the system:

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list log-update rate 10
```

# ipv4 access-list log-update threshold

To specify the number of updates that are logged for IPv4 access lists, use the **ipv4 access-list log-update threshold** command in XR Config mode. To return the number of logged updates to the default setting, use the **no** form of this command.

**ipv4 access-list log-update threshold** *update-number*
**no ipv4 access-list log-update threshold** *update-number*

## Syntax Description

| | |
|---|---|
| update-number | Number of updates that are logged for every IPv4 access list configured on the router. Range is 0 to 2147483647. |

## Command Default

For IPv4 access lists, 2147483647 updates are logged.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

IPv4 access list updates are logged at 5-minute intervals, following the first logged update. Configuring a lower number of updates (a number lower than the default) is useful when more frequent update logging is desired.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| acl | read, write |

## Examples

This example shows how to configure a log threshold of ten updates for every IPv4 access list configured on the router:

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list log-update threshold 10
```

## Related Commands

| Command | Description |
|---|---|
| deny (IPv4) , on page 13 | Sets the deny conditions for an IPv4 access list. |
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| permit (IPv4) , on page 38 | Sets the permit conditions for an IPv4 access list |
| show access-lists ipv4 , on page 63 | Displays the contents of all current IPv4 access lists. |

# ipv6 access-group

To control access to an interface, use the **ipv6 access-group** command in interface configuration mode. To remove the specified access group, use the **no** form of this command.

**ipv6 access-group** *access-list-name* {**ingress** | **egress**} [**interface-statistics**]

## Syntax Description

| | |
|---|---|
| access-list-name | Name of an IPv6 access list as specified by an **ipv6 access-list** command. |
| ingress | Filters on inbound packets. |
| egress | Filters on outbound packets. |
| interface-statistics | (Optional) Specifies per-interface statistics in the hardware. |

## Command Default

The interface does not have an IPv6 access list applied to it.

## Command Modes

Interface configuration

## Usage Guidelines

The **ipv6 access-group** command is similar to the **ipv4 access-group** command, except that it is IPv6-specific.

Use the **ipv6 access-group** command to control access to an interface. To remove the specified access group, use the **no** form of the command. Use the *access-list-name* to specify a particular IPv6 access list. Use the **ingress** keyword to filter on inbound packets or the **egress** keyword to filter on outbound packets.

**Note** For packet filtering applications using the **ipv6 access-group** command, packet counters are maintained in hardware for each direction. If an access group is used on multiple interfaces in the same direction, then packets are counted for each interface.

If the access list permits the addresses, the software continues to process the packet. If the access list denies the address, the software discards the packet and returns a rate-limited Internet Control Message Protocol (ICMP) host unreachable message.

If the specified access list does not exist, all packets are passed.

By default, the unique or per-interface ACL statistics are disabled.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| ipv6 | read, write |

## Examples

The following example shows how to apply filters on packets inbound and outbound from GigabitEthernet interface 0/2/0/2:

```
RP/0/

/CPU0:router(config)# interface gigabitethernet 0/2/0/2
RP/0/

/CPU0:router(config-if)# ipv6 access-group p-in-filter ingress
RP/0/

/CPU0:router(config-if)# ipv6 access-group p-out-filter egress
```

The following example shows how to apply per-interface statistics in the hardware:

```
RP/0/

/CPU0:router(config)# interface gigabitethernet 0/2/0/2
RP/0/

/CPU0:router(config-if)# ipv6 access-group p-in-filter ingress interface-statistics
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 access-list(BNG) | Defines an IPv6 access list and enters IPv6 access list configuration mode. |
| show ipv6 interface | Displays the usability status of interfaces configured for IPv6. |

# ipv6 access-list

To define an IPv6 access list and to place the router in IPv6 access list configuration mode, use the **ipv6 access-list** command in XR Config mode. To remove the access list, use the **no** form of this command.

**ipv6 access-list** *name*

**Syntax Description**

| | |
|---|---|
| *name* | Name of the IPv6 access list. Names cannot contain a space or quotation mark, or begin with a numeric. |

**Command Default**

No IPv6 access list is defined.

**Command Modes**

XR Config mode

**Usage Guidelines**

The **ipv6 access-list** command is similar to the **ipv4 access-list** command, except that it is IPv6-specific.

The IPv6 access lists are used for traffic filtering based on source and destination addresses, IPv6 option headers, and optional, upper-layer protocol type information for finer granularity of control. IPv6 access lists are defined by using the **ipv6 access-list** command in global configuration mode and their permit and deny conditions are set by using the **deny** and **permit** commands in IPv6 access list configuration mode. Configuring the **ipv6 access-list** command places the router in IPv6 access list configuration mode—the router prompt changes to router (config-ipv6-acl)#. From IPv6 access list configuration mode, permit and deny conditions can be set for the defined IPv6 access list.

**Note** Every IPv6 access list has an implicit **deny ipv6 any any** statement as its last match condition. An IPv6 access list must contain at least one entry for the implicit **deny ipv6 any any** statement to take effect.

**Note** IPv6 prefix lists, not access lists, should be used for filtering routing protocol prefixes.

Use the **ipv6 access-group** interface configuration command with the *access-list-name* argument to apply an IPv6 access list to an IPv6 interface.

**Note** Every IPv6 ACL has implicit **permit icmp any any nd-na** , **permit icmp any any nd-ns** , and **deny ipv6 any any** statements as its last match conditions. (The former two match conditions allow for ICMPv6 neighbor discovery.) An IPv6 ACL must contain at least one entry for the implicit **deny ipv6 any any** statement to take effect. **permit icmp any any nd-na permit icmp any any nd-ns deny ipv6 any any deny ipv6 any any**

The IPv6 neighbor discovery process makes use of the IPv6 network layer service; therefore, by default, IPv6 ACLs implicitly allow IPv6 neighbor discovery packets to be sent and received on an interface. In IPv4, the Address Resolution Protocol (ARP), which is equivalent to the IPv6 neighbor discovery process, makes use of a separate data link layer protocol; therefore, by default, IPv4 ACLs implicitly allow ARP packets to be sent and received on an interface.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| ipv6 | read, write |

## Examples

The following example shows how to configure the IPv6 access list named list2 and applies the ACL to outbound traffic on interface GigabitEthernet 0/2/0/2. Specifically, the first ACL entry keeps all packets from the network fec0:0:0:2::/64 (packets that have the site-local prefix fec0:0:0:2 as the first 64 bits of their source IPv6 address) from exiting out of interface GigabitEthernet 0/2/0/2. The second entry in the ACL permits all other traffic to exit out of interface GigabitEthernet 0/2/0/2. The second entry is necessary because an implicit deny all condition is at the end of each IPv6 ACL.

```
RP/0/

/CPU0:router(config)# ipv6 access-list list2
RP/0/

/CPU0:router(config-ipv6-acl)# 10 deny fec0:0:0:2::/64 any
RP/0/

/CPU0:router(config-ipv6-acl)# 20 permit any any

RP/0/

/CPU0:router# show ipv6 access-lists list2

ipv6 access-list list2
  10 deny ipv6 fec0:0:0:2::/64 any
  20 permit ipv6 any any

RP/0/

/CPU0:router(config)# interface gigabitethernet 0/2/0/2
RP/0/

/CPU0:router(config-if)# ipv6 access-group list2 out
```

**Note** IPv6 is automatically configured as the protocol type in **permit any any** and **deny any any** statements that are translated from global configuration mode to IPv6 access list configuration mode.

**Note** An IPv6 router does not forward to another network an IPv6 packet that has a link-local address as either its source or destination address (and the source interface for the packet is different from the destination interface for the packet).

# ipv6 access-list log-update rate

To specify the rate at which IPv6 access lists are logged, use the **ipv6 access-list log-update rate** command in XR Config mode. To return the update rate to the default setting, use the **no** form of this command.

**ipv6 access-list log-update rate** *rate-number*
**no ipv6 access-list log-update rate** *rate-number*

## Syntax Description

| | |
|---|---|
| rate-number | Rate at which IPv6 access hit logs are generated per second on the router. Range is 1 to 1000. |

## Command Default

Default is 1.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The *rate-number* argument applies to all the IPv6 access-lists configured on the interfaces. That is, at any given time there can be between 1 and 1000 log entries for the system.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| acl | read, write |

## Examples

This example shows how to configure a IPv6 access hit logging rate for the system:

```
RP/0/RP0/CPU0:router(config)# ipv6 access-list log-update rate 10
```

# ipv6 access-list log-update threshold

To specify the number of updates that are logged for IPv6 access lists (ACLs), use the **ipv6 access-list log-update threshold** command in XR Config mode. To return the number of logged updates to the default setting, use the **no** form of this command.

**ipv6 access-list log-update threshold** *update-number*
**no ipv6 access-list log-update threshold** *update-number*

## Syntax Description

| | |
|---|---|
| update-number | Number of updates that are logged for every IPv6 access list configured on the router. Range is 0 to 2147483647. |

## Command Default

For IPv6 access lists, 350000 updates are logged.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **ipv6 access-list log-update threshold** command is similar to the **ipv4 access-list log-update threshold** command, except that it is IPv6-specific.

IPv6 access list updates are logged at 5-minute intervals, following the first logged update. Configuring a lower number of updates (a number lower than the default) is useful when more frequent update logging is desired.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| ipv6 | read, write |

## Examples

This example shows how to configure a log threshold of ten updates for every IPv6 access list configured on the router:

```
RP/0/RP0/CPU0:router(config)# ipv6 access-list log-update threshold 10
```

# permit (IPv4)

To set conditions for an IPv4 access list, use the **permit** command in access list configuration mode. There are two versions of the **permit** command: **permit** (source), and **permit** (protocol). To remove a condition from an access list, use the **no** form of this command.

[*sequence-number*] **permit** *source* [*source-wildcard*] [{**log** | **log-input**}]
**no** *sequence-number*

**Internet Control Message Protocol (ICMP)**
[*sequence-number*] **permit icmp** *source source-wildcard destination destination-wildcard* [*icmp-type*] [*icmp-code*] [**precedence** *precedence*] [**dscp** *dscp*] [**fragments**] [{**log** | **log-input**}] [**icmp-off**]

**Internet Group Management Protocol (IGMP)**
[*sequence-number*] **permit igmp** *source source-wildcard destination destination-wildcard* [*igmp-type*] [**precedence** *precedence*] [**dscp** *value*] [**fragments**] [{**log** | **log-input**}]

**User Datagram Protocol (UDP)**
[*sequence-number*] **permit udp** *source source-wildcard* [*operator* {*portprotocol-port*}] *destination destination-wildcard* [*operator* {*portprotocol-port*}] [**precedence** *precedence*] [**dscp** *dscp*] [**fragments**] [{**log** | **log-input**}]

## Syntax Description

| | |
|---|---|
| **default** | (Optional) Specifies the default next hop for this entry.<br>If the **default** keyword is configured, ACL-based forwarding action is taken only if the results of the PLU lookup for the destination of the packets determine a default route; that is, no specified route is determined to the destination of the packet. |
| **capture** | Captures matching traffic.<br>When the acl command is configured on the source mirroring port, if the ACL configuration command does not use the **capture** keyword, no traffic gets mirrored. If the ACL configuration uses the **capture** keyword, but the acl command is not configured on the source port, then the whole port traffic is mirrored and the **capture** action does not have any affect. |

| | |
|---|---|
| *ipv4-address1 ipv4-address2 ipv4-address3* | (Optional) Uses one to three next-hop addresses. The IP address types are defined as follows:<br>• Default IP addresses—Specifies the next-hop router in the path toward the destination in which the packets must be forwarded, if there is no explicit route for the destination address of the packet in the routing table. The first IP address that is associated with a connected interface that is currently up is used to route the packets.<br>• Specified IP addresses—Specifies the next-hop router in the path toward the destination in which the packets must be forwarded. The first IP address that is associated with a connected interface that is currently up is used to route the packets. |

| | |
|---|---|
| **dscp** *dscp* | (Optional) Differentiated services code point (DSCP) provides quality of service control. The values for *dscp* are as follows:<br>• 0–63—Differentiated services codepoint value<br>• af11—Match packets with AF11 dscp (001010)<br>• af12—Match packets with AF12 dscp (001100)<br>• af13—Match packets with AF13 dscp (001110)<br>• af21—Match packets with AF21 dscp (010010)<br>• af22—Match packets with AF22 dscp (010100)<br>• af23—Match packets with AF23 dscp (010110)<br>• af31—Match packets with AF31 dscp (011010)<br>• af32—Match packets with AF32 dscp (011100)<br>• af33—Match packets with AF33 dscp (011110)<br>• af41—Match packets with AF41 dscp (100010)<br>• af42—Match packets with AF42 dscp (100100)<br>• af43–Match packets with AF43 dscp (100110)<br>• cs1—Match packets with CS1 (precedence 1) dscp (001000)<br>• cs2—Match packets with CS2 (precedence 2) dscp (010000)<br>• cs3—Match packets with CS3 (precedence 3) dscp (011000)<br>• cs4—Match packets with CS4 (precedence 4) dscp (100000)<br>• cs5—Match packets with CS5 (precedence 5) dscp (101000)<br>• cs6—Match packets with CS6 (precedence 6) dscp (110000)<br>• cs7—Match packets with CS7 (precedence 7) dscp (111000)<br>• default—Default DSCP (000000)<br>• ef—Match packets with EF dscp (101110) |

| | |
|---|---|
| fragments | (Optional) Causes the software to examine noninitial fragments of IPv4 packets when applying this access list entry. When this keyword is specified, fragments are subject to the access list entry. |
| log | (Optional) Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.)<br>The message includes the access list number, whether the packet was permitted or denied; the protocol, whether it was TCP, UDP, ICMP, or a number; and, if appropriate, the source and destination addresses and source and destination port numbers. The message is generated for the first packet that matches a flow, and then at 5-minute intervals, including the number of packets permitted or denied in the prior 5-minute interval. |
| log-input | (Optional) Provides the same function as the **log** keyword, except that the logging message also includes the input interface. |
| nexthop1, nexthop2, nexthop3 | (Optional) Forwards the specified next hop for this entry. |
| ttl | (Optional) Turns on matching against time-to-life (TTL) value. |
| *ttl value* [*value1 ... value2*] | (Optional) TTL value used for filtering. Range is 1 to 255.<br>If only *value* is specified, the match is against this value.<br>If both *value1* and *value2* are specified, the packet TTL is matched against the range of TTLs between *value1* and *value2* . |
| icmp-off | (Optional) Turns off ICMP generation for denied packets |

| | |
|---|---|
| icmp-type | (Optional) ICMP message type for filtering ICMP packets. Range is from 0 to 255. |
| icmp-code | (Optional) ICMP message code for filtering ICMP packets. Range is from 0 to 255. |
| igmp-type | (Optional) IGMP message type (0 to 15) or message name for filtering IGMP packets, as follows:<br>• dvmrp<br>• host-query<br>• host-report<br>• mtrace<br>• mtrace-response<br>• pim<br>• precedence<br>• trace<br>• v2-leave<br>• v2-report<br>• v3-report |
| operator | (Optional) Operator is used to compare source or destination ports. Possible operands are **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range).<br>If the operator is positioned after the *source* and *source-wildcard* values, it must match the source port.<br>If the operator is positioned after the *destination* and *destination-wildcard* values, it must match the destination port.<br>If the operator is positioned after the **ttl** keyword, it matches the TTL value.<br>The **range** operator requires two port numbers. All other operators require one port number. |

| | |
|---|---|
| port | Decimal number a TCP or UDP port. Range is 0 to 65535.<br>TCP ports can be used only when filtering TCP. UDP ports can be used only when filtering UDP. |
| protocol-port | Name of a TCP or UDP port. TCP and UDP port names are listed in the "Usage Guidelines" section.<br>TCP port names can be used only when filtering TCP. UDP port names can be used only when filtering UDP. |
| established | (Optional) For the TCP protocol only: Indicates an established connection. |
| match-any | (Optional) For the TCP protocol only: Filters on any combination of TCP flags. |
| match-all | (Optional) For the TCP protocol only: Filters on all TCP flags. |
| + \| - | (Required) For the TCP protocol **match-any** , **match-all** : Prefix *flag-name* with + or - . Use the + *flag-name* argument to match packets with the TCP flag set. Use the - *flag-name* argument to match packets when the TCP flag is not set. |
| flag-name | (Optional) For the TCP protocol **match-any** , **match-all** . Flag names are: **ack** , **fin** , **psh** , **rst** , **syn** . |
| **counter** | (Optional) Enables accessing ACL counters using SNMP query. The **counter** *counter-name* keyword is available on Cisco ASR 9000 Enhanced Ethernet Line Cards only. |
| *counter-name* | Defines an ACL counter name. |

**Command Default**

There is no specific condition under which a packet is denied passing the IPv4 access list.

ICMP message generation is enabled by default.

**Command Modes**

IPv4 access list configuration

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |
| Release 5.2.1 | The **default nexthop** and **nexthop** keywords were added to support ACL-based forwarding. |

**Usage Guidelines**

Use the **permit** command following the **ipv4 access-list** command to specify conditions under which a packet can pass the access list.

By default, the first statement in an access list is number 10, and the subsequent statements are incremented by 10.

You can add **permit**, **deny**, or **remark** statements to an existing access list without retyping the entire list. To add a new **s**tatement anywhere other than at the end of the list, create a new statement with an appropriate entry number that falls between two existing entry numbers to indicate where it belongs.

If you want to add a statement between two consecutively numbered statements (for example, between lines 10 and 11), first use the **resequence access-list** command to renumber the first statement and increment the entry number of each subsequent statement. The *increment* argument causes new, unused line numbers between statements. Then add a new statement with the *entry-number* specifying where it belongs in the access list.

The following is a list of precedence names:

- critical
- flash
- flash-override
- immediate
- internet
- network
- priority
- routine

The following is a list of ICMP message type names:

- administratively-prohibited
- alternate-address
- conversion-error
- dod-host-prohibited
- dod-net-prohibited
- echo
- echo-reply
- general-parameter-problem
- host-isolated
- host-precedence-unreachable
- host-redirect
- host-tos-redirect
- host-tos-unreachable
- host-unknown

- host-unreachable
- information-reply
- information-request
- mask-reply
- mask-request
- mobile-redirect
- net-redirect
- net-tos-redirect
- net-tos-unreachable
- net-unreachable
- network-unknown
- no-room-for-option
- option-missing
- packet-too-big
- parameter-problem
- port-unreachable
- precedence-unreachable
- protocol-unreachable
- reassembly-timeout
- redirect
- router-advertisement
- router-solicitation
- source-quench
- source-route-failed
- time-exceeded
- timestamp-reply
- timestamp-request
- traceroute
- ttl-exceeded
- unreachable

The following is a list of TCP port names that can be used instead of port numbers. Refer to the current *Assigned Numbers* RFC to find a reference to these protocols. You can find port numbers corresponding to these protocols by typing a ? in the place of a port number.

- bgp
- chargen
- cmd
- daytime
- discard
- domain
- echo
- exec
- finger
- ftp
- ftp-data
- gopher

- hostname
- ident
- irc
- klogin
- kshell
- login
- lpd
- nntp
- pim-auto-rp
- pop2
- pop3
- smtp
- sunrpc
- tacacs
- talk
- telnet
- time
- uucp
- whois
- www

The following UDP port names can be used instead of port numbers. Refer to the current *Assigned Numbers* RFC to find a reference to these protocols. You can find port numbers corresponding to these protocols by typing a ? in the place of a port number.

- biff
- bootpc
- bootps
- discard
- dnsix
- domain
- echo
- isakmp
- mobile-ip
- nameserver
- netbios-dgm
- netbios-ns
- netbios-ss
- ntp
- pim-auto-rp
- rip
- snmp
- snmptrap
- sunrpc
- syslog
- tacacs
- talk

- tftp
- time
- who
- xdmcp

Use the following flags in conjunction with the **match-any** and **match-all** keywords and the + and - signs to select the flags to display:

- ack
- fin
- psh
- rst
- syn

For example, **match-all** *+ack +syn* displays TCP packets with both the ack *and* syn flags set, or **match-any** *+ack syn* displays the TCP packets with the ack set *or* the syn not set.

For ACL-based forwarding, we recommend that you use the **permit** command and **any any** keywords for the last ACL-based forwarding ACE rule to overwrite an implicit deny of security ACL. It ensures that all packets are forwarded with the traditional destination IP address if you do not want to drop any non-ABF related packets.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| acl | read, write |

## Examples

The following example shows how to set a permit condition for an access list named Internetfilter:

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list Internetfilter
RP/0/RP0/CPU0:router(config-ipv4-acl)# 10 permit 192.168.34.0 0.0.0.255
RP/0/RP0/CPU0:router(config-ipv4-acl)# 20 permit 172.16.0.0 0.0.255.255
RP/0/RP0/CPU0:router(config-ipv4-acl)# 25 permit tcp host 172.16.0.0 eq bgp host
192.168.202.203 range 1300 1400
RP/0/RP0/CPU0:router(config-ipv4-acl)# deny 10.0.0.0 0.255.255.255
```

## Related Commands

| Command | Description |
|---|---|
| deny (IPv4) , on page 13 | Sets the conditions for an IPv4 access list. |
| ipv4 access-group, on page 27 | Filters incoming or outgoing IPv4 traffic on an interface. |

| Command | Description |
|---|---|
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| remark (IPv4) , on page 54 | Inserts a helpful remark about an IPv4 access list entry. |
| resequence access-list ipv4 , on page 58 | Changes the starting entry number of the first statement in an existing IPv4 access list, and the number by which subsequent statements are incremented. |
| show access-lists ipv4 , on page 63 | Displays the contents of all current IPv4 access lists. |

# permit (IPv6)

To set permit conditions for an IPv6 access list, use the **permit** command in IPv6 access list configuration mode. To remove the permit conditions, use the **no** form of this command.

**no** *sequence-number*

**Internet Control Message Protocol (ICMP)**
*[ sequence-number]* **permit icmp** [*icmp-type*] [ *icmp-code*] [**dscp** *value*] [ routing] [**authen**] [**destopts**] [ **fragments**] [ **log**] [**log-input**] [**icmp-off**]

**Transmission Control Protocol (TCP)**
*[sequence-number]* **permit tcp** [*operator*{*port* | *protocol-port*}] [*operator*{*port* | *protocol* | *port*}] [**dscp** *value*] [**routing**] [**authen**] [**destopts**] [**fragments**] [**established**] {**match-any | match-all | + | -**} [*flag-name*] [**log**] [**log-input**]

**User Datagram Protocol (UDP)**
*[sequence-number]* **permit tcp** [*operator*{*port* | *protocol-port*}] [*operator*{*port* | *protocol* | *port*}] [**dscp** *value*] [**routing**] [**authen**] [**destopts**] [**fragments**] [**established**] [*flag-name*] [**log**] [**log-input**]

## Syntax Description

| | |
|---|---|
| sequence-number | (Optional) Number of the **permit** statement in the access list. This number determines the order of the statements in the access list. Range is 1 to 2147483646. (By default, the first statement is number 10, and the subsequent statements are incremented by 10.) Use the **resequence access-list** command to change the number of the first statement and increment subsequent statements of a configured access list. |
| protocol | Name or number of an Internet protocol. It can be one of the keywords **ahp** , **esp** , **icmp** , **ipv6** , **pcp** , **sctp** , **tcp** , or **udp** , or an integer in the range from 0 to 255 representing an IPv6 protocol number. |
| *source-ipv6-prefix* / *prefix-length* | Source IPv6 network or class of networks about which to set permit conditions.<br>This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| any | An abbreviation for the IPv6 prefix ::/0. |
| **default** | (Optional) Specifies the default next hop for this entry.<br>If the **default** keyword is configured, ACL-based forwarding action is taken only if the results of the PLU lookup for the destination of the packets determine a default route; that is, no specified route is determined to the destination of the packet. |
| nexthop1, nexthop2, nexthop3 | (Optional) Forwards the specified next hop for this entry. |
| **host** *source-ipv6-address* | Source IPv6 host address about which to set permit conditions.<br>This *source-ipv6-address* argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |

| | |
|---|---|
| *operator* {*port* \| *protocol-port*} | (Optional) Operand that compares the source or destination ports of the specified protocol. Operands are **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range).<br>If the operator is positioned after the *source-ipv6-prefix / prefix-length* argument, it must match the source port.<br>If the operator is positioned after the *destination-ipv6-prefix / prefix-length* argument, it must match the destination port.<br>The **range** operator requires two port numbers. All other operators require one port number.<br>The *port* argument is the decimal number of a TCP or UDP port. A port number is a number from 0 to 65535. The *protocol-port* argument is the name of a TCP or UDP port. TCP port names can be used only when filtering TCP. UDP port names can be used only when filtering UDP. |
| *destination-ipv6-prefix / prefix-length* | Destination IPv6 network or class of networks about which to set permit conditions.<br>This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **host** *destination-ipv6-address* | Specifies the destination IPv6 host address about which to set permit conditions.<br>This *destination-ipv6-address* argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **dscp** *value* | (Optional) Matches a differentiated services code point (DSCP) value against the traffic class value in the Traffic Class field of each IPv6 packet header. Range is 0 to 63. |
| routing | (Optional) Matches source-routed packets against the routing extension header within each IPv6 packet header. |
| authen | (Optional) Matches if the IPv6 authentication header is present. |
| destopts | (Optional) Matches if the IPv6 destination options header is present. |
| fragments | (Optional) Matches non-initial fragmented packets where the fragment extension header contains a nonzero fragment offset. The **fragments** keyword is an option only if the *operator* [ *port-number* ] arguments are not specified. |
| packet-length operator | (Optional) Packet length operator used for filtering. |
| packet-length value | (Optional) Packet length used to match only packets in the range of the length. |

| | |
|---|---|
| log | (Optional) Causes an informational logging message about the packet that matches the entry to be sent to the console. (The level of messages logged to the console is controlled by the **logging console** command.)<br>The message includes the access list name and sequence number, whether the packet was permitted; the protocol, whether it was TCP, UDP, ICMP, or a number; and, if appropriate, the source and destination addresses and source and destination port numbers. The message is generated for the first packet that matches, and then at 5-minute intervals, including the number of packets permitted in the prior 5-minute interval. |
| log-input | (Optional) Provides the same function as the **log** keyword, except that the logging message also includes the input interface. |
| ttl | (Optional) Turns on matching against time-to-life (TTL) value. |
| operator | (Optional) Operand that compares the source or destination ports of the specified protocol. Operands are **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range** (inclusive range). |
| ttl value1 value2 | (Optional) TTL value used for filtering. Range is 1 to 255.<br>If only *value1* is specified, the match is against this value.<br>If both *value1* and *value2* are specified, the packet TTL is matched against the range of TTLs between *value1* and *value2* . |
| icmp-off | (Optional) Turns off ICMP generation for denied packets |
| icmp-type | (Optional) ICMP message type for filtering ICMP packets. Range is from 0 to 255. |
| icmp-code | (Optional) ICMP message code for filtering ICMP packets. Range is from 0 to 255. |
| established | (Optional) For the TCP protocol only: Indicates an established connection. |
| match-any | (Optional) For the TCP protocol only: Filters on any combination of TCP flags. |
| match-all | (Optional) For the TCP protocol only: Filters on all TCP flags. |
| + \| - | (Required) For the TCP protocol **match-any** , **match-all** : Prefix *flag-name* with + or - . Use the + *flag-name* argument to match packets with the TCP flag set. Use the - *flag-name* argument to match packets when the TCP flag is not set. |
| flag-name | (Required) For the TCP protocol **match-any** , **match-all** . Flag names are: ack, fin, psh, rst, syn. |

**Command Default**

No IPv6 access list is defined.

ICMP message generation is enabled by default.

**Command Modes**

IPv6 access list configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |
| Release 5.2.1 | The **default nexthop** and **nexthop** keywords were added to support ACL-based forwarding. |

## Usage Guidelines

The **permit** (IPv6) command is similar to the **permit** (IPv4) command, except that it is IPv6-specific.

Use the **permit** (IPv6) command following the **ipv6 access-list** command to define the conditions under which a packet passes the access list.

Specifying **ipv6** for the *protocol* argument matches against the IPv6 header of the packet.

By default, the first statement in an access list is number 10, and the subsequent statements are numbered in increments of 10.

You can add **permit**, **deny, or remark** statements to an existing access list without retyping the entire list. To add a new statement anywhere other than the end of the list, create a new statement with an appropriate entry number that falls between two existing entry numbers to indicate where it belongs.

Both the *source-ipv6-prefix/prefix-length* and *destination-ipv6-prefix/prefix-length* arguments are used for traffic filtering (the source prefix filters traffic based upon the traffic source; the destination prefix filters traffic based upon the traffic destination).

**Note** IPv6 prefix lists, and not access lists, should be used for filtering routing protocol prefixes.

The **fragments** keyword is an option available only if the *operator* [*port* | *protocol-port*] arguments are not specified.

**Note** If any ACE in an ACL contains ABF clause, this ACL cannot be applied at any non-zero compression level.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

This example shows how to configure the IPv6 access list named toCISCO and applies the access list to outbound traffic on HundredGigE interface 0/2/0/2. Specifically, the first deny entry in the list keeps all packets that have a destination TCP port number greater than 5000 from exiting out of HundredGigE interface 0/2/0/2. The second deny entry in the list keeps all packets that have a source UDP port number less than 5000 from exiting out of HundredGigE interface 0/2/0/2. The second deny entry also logs all matches to the console. The first permit entry in the list permits all ICMP packets to exit out of HundredGigE interface 0/2/0/2. The second permit entry in the list permits all other traffic to exit out of HundredGigE interface 0/2/0/2. The second permit entry is necessary because an implicit deny all condition is at the end of each IPv6 access list.

```
RP/0/RP0/CPU0:router(config)# ipv6 access-list toCISCO
RP/0/RP0/CPU0:router(config-ipv6-acl)# deny tcp any any gt 5000
RP/0/RP0/CPU0:router(config-ipv6-acl)# deny ipv6 any lt 5000 any log
RP/0/RP0/CPU0:router(config-ipv6-acl)# permit icmp any any
RP/0/RP0/CPU0:router(config-ipv6-acl)# permit any any
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/2/0/2
RP/0/RP0/CPU0:router(config-if)# ipv6 access-group tOCISCO out
```

```
RP/0/RP0/CPU0:router(config)# ipv6 access-list  v6-abf-acl
RP/0/RP0/CPU0:router(config-ipv6-acl)# 10 permit ipv6 any any default nexthop1 ipv6 11::1
nexthop2 ipv6 22::2 nexthop3 ipv6 33::3
RP/0/RP0/CPU0:router(config-ipv6-acl)# 20 permit ipv4 any any
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/0/2/0
RP/0/RP0/CPU0:router(config-if)#  ipv6 access-group v6-abf-acl ingress
```

## Related Commands

| Command | Description |
|---|---|
| deny (IPv6) , on page 22 | Sets deny conditions for an IPv6 access list. |
| ipv6 access-list, on page 34 | Defines an IPv6 access list and enters IPv6 access list configuration mode. |
| remark (IPv6) , on page 56 | Inserts a helpful remark about an IPv6 access list entry. |
| resequence access-list ipv6 , on page 60 | Changes the starting entry number of the first statement in an existing IPv6 access list, and the number by which subsequent statements are incremented. |

# remark (IPv4)

To write a helpful comment (remark) for an entry in an IPv4 access list, use the **remark** command in IPv4 access list configuration mode. To remove the remark, use the **no** form of this command.

[*sequence-number*] **remark** *remark*
**no** *sequence-number*

## Syntax Description

| | |
|---|---|
| sequence-number | (Optional) Number of the **remark** statement in the access list. This number determines the order of the statements in the access list. Range is 1 to 2147483646. (By default, the first statement is number 10; subsequent statements are incremented by 10.) |
| remark | Comment that describes the entry in the access list, up to 255 characters long. |

## Command Default

The IPv4 access list entries have no remarks.

## Command Modes

IPv4 access list configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **remark** command to write a helpful comment for an entry in an IPv4 access list. To remove the remark, use the **no** form of this command.

The remark can be up to 255 characters; anything longer is truncated.

If you know the sequence number of the remark you want to delete, you can remove it by entering the **no** *sequence-number* command.

Use the **resequence access-list ipv4** command if you want to add statements to an existing access list and the sequence numbers of consecutive entries do not permit additional statements.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| acl | read, write |

## Examples

In the following example, the user1 subnet is not allowed to use outbound Telnet:

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list telnetting
RP/0/RP0/CPU0:router(config-ipv4-acl)# 10 remark Do not allow user1 to telnet out
RP/0/RP0/CPU0:router(config-ipv4-acl)# 20 deny tcp host 172.16.2.88 255.255.0.0 any eq
telnet
```

```
RP/0/RP0/CPU0:router(config-ipv4-acl)# 30 permit icmp any any
RP/0/RP0/CPU0:router# show ipv4 access-list telnetting

ipv4 access-list telnetting
  0 remark Do not allow user1 to telnet out
  20 deny tcp 172.16.2.88 255.255.0.0 any eq telnet out
  30 permit icmp any any
```

## Related Commands

| Command | Description |
|---|---|
| deny (IPv4) , on page 13 | Sets the deny conditions for an IPv4 access list. |
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| permit (IPv4) , on page 38 | Sets the permit conditions for an IPv4 access list |
| resequence access-list ipv4 , on page 58 | Changes the starting entry number of the first statement in an existing IPv4 access list, and the number by which subsequent statements are incremented. |
| show access-lists ipv4 , on page 63 | Displays the contents of all current IPv4 access lists. |

# remark (IPv6)

To write a helpful comment (remark) for an entry in an IPv6 access list, use the **remark** command in IPv6 access list configuration mode. To remove the remark, use the **no** form of this command.

[*sequence-number*] **remark** *remark*
**no** *sequence-number*

## Syntax Description

| | |
|---|---|
| sequence-number | (Optional) Number of the **remark** statement in the access list. This number determines the order of the statements in the access list. Range is 1 to 2147483646. (By default, the first statement is number 10, and the subsequent statements are incremented by 10.) |
| remark | Comment that describes the entry in the access list, up to 255 characters long. |

## Command Default

The IPv6 access list entries have no remarks.

## Command Modes

IPv6 access list configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **remark** (IPv6) command is similar to the **remark** (IPv4) command, except that it is IPv6-specific.

Use the **remark** command to write a helpful comment for an entry in an IPv6 access list. To remove the remark, use the **no** form of this command.

The remark can be up to 255 characters; anything longer is truncated.

If you know the sequence number of the remark you want to delete, you can remove it by entering the **no** *sequence-number* command.

Use the **resequence access-list ipv6** command if you want to add statements to an existing access list and the sequence numbers of consecutive entries do not permit additional statements.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

In this example, a remark is added:

```
RP/0/RP0/CPU0:router(config)# ipv6 access-list Internetfilter
RP/0/RP0/CPU0:router(config-ipv6-acl)# 10 permit ipv6 3333:1:2:3::/64 any
RP/0/RP0/CPU0:router(config-ipv6-acl)# 20 permit ipv6 4444:1:2:3::/64 any
RP/0/RP0/CPU0:router(config-ipv6-acl)# 30 permit ipv6 5555:1:2:3::/64 any
RP/0/RP0/CPU0:router(config-ipv6-acl)# 39 remark Block BGP traffic from a given host
RP/0/RP0/CPU0:router(config-ipv6-acl)# 40 deny tcp host 6666:1:2:3::10 eq bgp host
```

```
7777:1:2:3::20 range 1300 1400
RP/0/RP0/CPU0:router# show ipv6 access-list Internetfilter

ipv6 access-list Internetfilter
  10 permit ipv6 3333:1:2:3::/64 any
  20 permit ipv6 4444:1:2:3::/64 any
  30 permit ipv6 5555:1:2:3::/64 any
  39 remark Block BGP traffic from a given host
  40 deny tcp host 6666:1:2:3::10 eq bgp host 7777:1:2:3::20 range host 6666:1:2:3::10 eq
bgp host 7777:1:2:3::20 range 1300 1400
```

## Related Commands

| Command | Description |
|---|---|
| deny (IPv6) , on page 22 | Sets the deny conditions for an IPv6 access list. |
| ipv6 access-list, on page 34 | Defines an IPv6 access list and enters IPv6 access list configuration mode. |
| permit (IPv6) , on page 49 | Sets permit conditions for an IPv6 access list |
| resequence access-list ipv6 , on page 60 | Changes the starting entry number of the first statement in an existing IPv6 access list, and the number by which subsequent statements are incremented. |

# resequence access-list ipv4

To renumber existing statements and increment subsequent statements to allow a new IPv4 access list statement (**permit**, **deny**, or **remark**) to be added, use the **resequence access-list ipv4** command in XR EXEC mode.

**resequence access-list ipv4** *name* [*base* [*increment*]]

## Syntax Description

| | |
|---|---|
| name | Name of an IPv4 access list. |
| base | (Optional) Number of the first statement in the specified access list, which determines its order in the access list. Maximum value is 2147483644. Default is 10. |
| increment | (Optional) Number by which the base sequence number is incremented for subsequent statements. Maximum value is 2147483644. Default is 10. |

## Command Default

*base*: 10

*increment*: 10

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **resequence access-list ipv4** command to add a **permit**, **deny**, or **remark** statement between consecutive entries in an existing IPv4 access list. Specify the first entry number (the *base*) and the increment by which to separate the entry numbers of the statements. The software renumbers the existing statements, thereby making room to add new statements with the unused entry numbers.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

In this example, suppose you have an existing access list:

```
ipv4 access-list marketing
  1 permit 10.1.1.1
  2 permit 10.2.0.0 0.0.255.255
  3 permit tcp host 10.2.2.2 255.255.0.0 any eq telnet
```

You want to add additional entries in the access list. First you resequence the entries, renumbering the statements starting with number 20 and an increment of 5, and then you have room for four additional statements between each of the existing statements:

```
RP/0/RP0/CPU0:router# resequence access-list ipv4 marketing 20 5
RP/0/RP0/CPU0:router# show access-lists ipv4 marketing
```

```
ipv4 access-list marketing
  20 permit 10.1.1.1
  25 permit 10.2.0.0
  30 permit tcp host 10.2.2.2 255.255.0.0 any eq telnet
```

Now you add your new entries.

```
RP/0/RP0/CPU0:router(config)# ipv4 access-list marketing
RP/0/RP0/CPU0:router(config-ipv4-acl)# 3 remark Do not allow user1 to telnet out
RP/0/RP0/CPU0:router(config-ipv4-acl)# 4 deny tcp host 172.16.2.88 255.255.0.0 any eq telnet
RP/0/RP0/CPU0:router(config-ipv4-acl)# 29 remark Allow user2 to telnet out
RP/0/RP0/CPU0:router# show access-lists ipv4 marketing

ipv4 access-list marketing
  3 remark Do not allow user1 to telnet out
  4 deny tcp host 171.69.2.88 255.255.0.0 any eq telnet
  20 permit 10.1.1.1
  25 permit 10.2.0.0
  29 remark Allow user2 to telnet out
  30 permit tcp host 10.2.2.2 255.255.0.0 any eq telnet
```

## Related Commands

| Command | Description |
|---|---|
| deny (IPv4) , on page 13 | Sets the deny conditions for an IPv4 access list. |
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| permit (IPv4) , on page 38 | Sets the permit conditions for an IPv4 access list |
| remark (IPv4) , on page 54 | Inserts a helpful remark about an IPv4 access list |
| show access-lists ipv4 , on page 63 | Displays the contents of all current IPv4 access lists. |

# resequence access-list ipv6

To renumber existing statements and increment subsequent statements to allow a new IPv6 access list statement (**permit**, **deny**, or **remark**) to be added, use the **resequence access-list ipv6** command in XR EXEC mode.

**resequence access-list ipv6** *name* [*base* [*increment*]]

## Syntax Description

| | |
|---|---|
| name | Name of an IPv6 access list. |
| base | (Optional) Number of the first statement in the specified access list, which determines its order in the access list. Maximum value is 2147483646. Default is 10. |
| increment | (Optional) Number by which the base sequence number is incremented for subsequent statements. Maximum value is 2147483644. Default is 10. |

## Command Default

*base*: 10

*increment*: 10

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **resequence access-list ipv6** command is similar to the **resequence access-list ipv4** command, except that it is IPv6 specific.

Use the **resequence access-list ipv6** command to add a **permit**, **deny**, or **remark** statement between consecutive entries in an existing IPv6 access list. Specify the first entry number (the *base*) and the increment by which to separate the entry numbers of the statements. The software renumbers the existing statements, thereby making room to add new statements with the unused entry numbers.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

In the following example, suppose you have an existing access list:

```
ipv6 access-list Internetfilter
  10 permit ipv6 3333:1:2:3::/64 any
  20 permit ipv6 4444:1:2:3::/64 any
  30 permit ipv6 5555:1:2:3::/64 any
```

You want to add additional entries in the access list. First, you resequence the entries, renumbering the statements starting with number 20 and an increment of 5, and then you have room for four additional statements between each of the existing statements:

```
RP/0/RP0/CPU0:router# resequence access-list ipv6 Internetfilter 20 5
RP/0/RP0/CPU0:router# show access-lists ipv6 Internetfilter

ipv6 access-list Internetfilter
  20 permit ipv6 3333:1:2:3::/64 any
  25 permit ipv6 4444:1:2:3::/64 any
  30 permit ipv6 5555:1:2:3::/64 any
```

Now you add your new entries.

```
RP/0/RP0/CPU0:router(config)# ipv6 access-list Internetfilter
RP/0/RP0/CPU0:router(config-ipv6-acl)# 3 remark Block BGP traffic from a given host
RP/0/RP0/CPU0:router(config-ipv6-acl)# 4 deny tcp host 6666:1:2:3::10 eq bgp host
7777:1:2:3::20 range 1300 1400
RP/0/RP0/CPU0:router# show access-lists ipv6 Internetfilter

ipv6 access-list Internetfilter
  3 remark Block BGP traffic from a given host
  4 deny tcp host 6666:1:2:3::10 eq bgp host 7777:1:2:3::20 range 1300 1404 deny tcp host
171.69.2.88 255.255.0.0 any eq telnet
  20 permit ipv6 3333:1:2:3::/64 any
  25 permit ipv6 4444:1:2:3::/64 any
  30 permit ipv6 5555:1:2:3::/64 any
```

## Related Commands

| Command | Description |
|---|---|
| deny (IPv6) , on page 22 | Sets the deny conditions for an IPv6 access list. |
| ipv6 access-list, on page 34 | Defines an IPv6 access list and enters IPv6 access list configuration mode. |
| permit (IPv6) , on page 49 | Set permit conditions for an IPv6 access list. |
| remark (IPv6) , on page 56 | Inserts a helpful remark about an IPv6 access list entry. |

# show access-lists afi-all

To display the contents of current IPv4 and IPv6 access lists, use the **show access-lists afi-all** command in XR EXEC mode.

**show access-lists afi-all**

## Syntax Description

This command has no keywords or arguments.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

This sample output is from the **show access-lists afi-all** command:

```
RP/0/RP0/CPU0:router# show access-lists afi-all

ipv4 access-list crypto-1
 10 permit ipv4 65.21.21.0 0.0.0.255 65.6.6.0 0.0.0.255
 20 permit ipv4 192.168.241.0 0.0.0.255 192.168.65.0 0.0.0.255
```

# show access-lists ipv4

To display the contents of current IPv4 access lists, use the **show access-lists ipv4** command in XR EXEC mode.

**show access-lists ipv4** [{*access-list-name* **hardware** {**ingress**|**egress**} **interface** *type interface-path-id* {**sequence** *number*|**location** *node-id*}|**summary** [*access-list-name*]|*access-list-name* [*sequence-number*] | **maximum** [**usage pfilter** { **location** *node-id* | **all**}]}]

## Syntax Description

| | |
|---|---|
| access-list-name | (Optional) Name of a particular IPv4 access list. The name cannot contain spaces or quotation marks, but can include numbers. |
| hardware | (Optional) Identifies the access list as an access list for an interface. |
| ingress | (Optional) Specifies an inbound interface. |
| egress | (Optional) Specifies an outbound interface. |
| interface | (Optional) Displays interface statistics. |
| type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| interface-path-id | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **sequence** *number* | (Optional) Sequence number of a particular IPv4 access list. Range is 1 to 2147483644. |

| | |
|---|---|
| **location** *node-id* | (Optional) Location of a particular IPv4 access list. The *node-id* argument is entered in the *rack/slot/module* notation. |
| summary | (Optional) Displays a summary of all current IPv4 access lists. |
| sequence-number | (Optional) Sequence number of a particular IPv4 access list. Range is 1 to 2147483644. |
| maximum | (Optional) Displays the current maximum number of configurable IPv4 access control lists (ACLs) and access control entries (ACEs). |
| **detail interface** *type interface-path-id* | (Optional) Displays detailed configuration of the ternary content addressable memory (TCAM) manager module of this ACL on the specified interface. |
| usage | (Optional) Displays the usage of the access list on a given line card. |
| pfilter | (Optional) Displays the packet filtering usage for the specified line card. |
| all | (Optional) Displays the location of all the line cards. |

## Command Default

The default displays all IPv4 access lists.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |
| Release 5.2.1 | The show command was updated to display ACL-based forwarding information. |

## Usage Guidelines

Use the **show access-lists ipv4** command to display the contents of all IPv4 access lists. To display the contents of a specific IPv4 access list, use the *name* argument. Use the *sequence-number* argument to specify the sequence number of the access list.

Use the **hardware , ingress** or **egress** , and **location** keywords to display the access list hardware contents and counters for all interfaces that use the specified access list in a given direction (ingress or egress). To display the contents of a specific access list entry, use the **sequence** *number* keyword and argument. The access group for an interface must be configured using the **ipv4 access-group** command for access list hardware counters to be enabled.

Use the **show access-lists ipv4 summary** command to display a summary of all current IPv4 access lists. To display a summary of a specific IPv4 access list, use the *name* argument.

Use the **show access-lists ipv4 maximum detail** command to display the OOR details for IPv4 access lists. OOR limits the number of ACLs and ACEs that can be configured in the system. When the limit is reached, configuration of new ACLs or ACEs is rejected.

Use the **show access-list ipv4 usage** command to display a summary of all interfaces and access lists programmed on the specified line card.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

In the following example, the contents of all IPv4 access lists are displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv4

ipv4 access-list 101
  10 deny udp any any eq ntp
  20 permit tcp any any
  30 permit udp any any eq tftp
  40 permit icmp any any
  50 permit udp any any eq domain
ipv4 access-list Internetfilter
  10 permit tcp any 172.16.0.0 0.0.255.255 eq telnet
  20 deny tcp any any
  30 deny udp any 172.18.0.0 0.0.255.255 lt 1024
  40 deny ipv4 any any log
```

In the following example, the contents of an access list named Internetfilter are displayed to show an example of ACL-based forwarding:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 Internetfilter

ipv4 access-list Internetfilter
  10 permit ipv4 host 50.3.3.3 any nexthop 1.1.1.1 2.2.2.2 3.3.3.3
  20 permit ipv4 host 50.60.1.2 any nexthop 50.70.1.2 50.80.1.2
  25 permit ipv4 host 50.2.2.2 any nexthop 50.70.1.2
  30 permit ipv4 host 50.70.1.2 any nexthop 50.80.1.2
  40 permit ipv4 host 1.1.1.1 any nexthop 50.70.1.2
  50 permit ipv4 host any any
```

In the following example, the contents of an access list named acl_hw_1 are displayed to show an example of ACL-based forwarding for a specific access list entry for the hardware **detail** option:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 acl_hw_1 hardware ingress sequence 20 detail
location 0/1/CPU0

ACL name: ucode
Sequence Number: 20
Grant: permit
Logging: OFF
Per ace icmp: ON
Next Hop Enable: ON <<<<<<<<< (ABF specific)
Next-hop: 50.70.1.2 <<<<<<<<< (ABF specific)
Default Next Hop: OFF<<<<<<<<< (ABF specific)
```

```
Hits: 661765
Statistics pointer: 0x60016
Number of TCAM entries: 1


Entry : 0 for ACE : 20
RAW value  : 0x00000040 0xffffffff 0xffffff11 0x0000007f 0xfdf2ffff 0xffffffff
RAW mask   : 0x000000ff 0xfc000000 0x000000ff 0x00000080 0xffff0000 0000000000
RAW result : 0x00000000 0x00000003 0x00000000 0x01010101

--------------------------------Field Details---------------------------------
acl_id             : 0x03f
acl_id mask        : 0x3ff
```

In the following example, the contents of an access list named acl_hw_1 are displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 acl_hw_1 hardware egress location 0/2/cp0

ipv4 access-list acl_hw_1
  10 permit icmp 192.168.36.0 0.0.0.255 any (251 hw matches)
  20 permit ip 172.16.3.0 0.0.255.255 any (29 hw matches)
  30 deny tcp any any (58 hw matches)
```

This table describes the significant fields shown in the display.

***Table 2: show access-lists ipv4 hardware Field Descriptions***

| Field | Description |
|---|---|
| hw matches | Number of hardware matches. |
| ACL name | Name of the ACL programmed in hardware. |
| Sequence Number | Each ACE sequence number is programmed into hardware with all the fields that are corresponding to the values set in ACE. |
| Grant | Depending on the ACE rule, the grant is set to deny, permit, or both. |
| Logging | Logging is set to on if ACE uses a log option to enable logs. |
| Per ace icmp | If Per ace icmp is set to on in the hardware, ICMP is unreachable, is rate-limited, and is generated. The default is set to on. |
| Hits | Hardware counter for that ACE. |

In the following example, a summary of all IPv4 access lists are displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 summary

ACL Summary:
  Total ACLs configured: 3
  Total ACEs configured: 11
```

This table describes the significant fields shown in the display.

*Table 3: show access-lists ipv4 summary Field Descriptions*

| Field | Description |
|---|---|
| Total ACLs configured | Number of configured IPv4 ACLs. |
| Total ACEs configured | Number of configured IPV4 ACEs. |

In the following example, the OOR details of the IPv4 access lists are displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 maximum detail

Default max configurable acls :5000
Default max configurable aces :200000
Current configured acls       :1
Current configured aces       :2
Current max configurable acls :5000
Current max configurable aces :200000
Max configurable acls         :9000
Max configurable aces         :350000
```

This table describes the significant fields shown in the display.

*Table 4: show access-lists ipv4 maximum detail Field Descriptions*

| Field | Description |
|---|---|
| Default max configurable acls | Default maximum number of configurable IPv4 ACLs allowed. |
| Default max configurable aces | Default maximum number of configurable IPv4 ACEs allowed. |
| Current configured acls | Number of configured IPv4 ACLs. |
| Current configured aces | Number of configured IPv4 ACEs. |
| Current max configurable acls | Configured maximum number of configurable IPv4 ACLs allowed. |
| Current max configurable aces | Configured maximum number of configurable IPv4 ACEs allowed. |
| Max configurable acls | Maximum number of configurable IPv4 ACLs allowed. |
| Max configurable aces | Maximum number of configurable IPv4 ACEs allowed. |

## Related Commands

| Command | Description |
|---|---|
| clear access-list ipv4, on page 3 | Resets the IPv4 access list match counters. |
| copy access-list ipv4 , on page 9 | Copies an existing IPv4 access list. |
| deny (IPv4) , on page 13 | Sets the deny conditions for an ACE of an IPv4 access list. |
| ipv4 access-group, on page 27 | Filters incoming or outgoing IPv4 traffic on an interface. |

| Command | Description |
|---|---|
| ipv4 access-list, on page 29 | Defines an IPv4 access list and enters IPv4 access list configuration mode. |
| permit (IPv4) , on page 38 | Sets the permit conditions for an ACE of an IPv4 access list. |
| remark (IPv4) , on page 54 | Inserts a helpful remark about an IPv4 access list entry. |
| resequence access-list ipv4 , on page 58 | Changes the starting entry number of the first statement in an existing IPv4 access list, and the number by which subsequent statements are incremented. |

# show access-lists ipv4 standby

To display the contents of current IPv4 standby access lists, use the **show access-lists ipv4 standby** command in XR EXEC mode.

**show access-lists ipv4 standby** [*access-list-name*] [**summary**]

## Syntax Description

| | |
|---|---|
| access-list name | (Optional) Name of a particular IPv4 access list. The name cannot contain spaces or quotation marks, but can include numbers. |
| summary | (Optional) Displays a summary of all current IPv4 standby access lists. |

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced . |

## Usage Guidelines

Use the **show access-lists ipv4 standby** command to display the contents of current IPv4 standby access lists. To display the contents of a specific IPv4 access list, use the *name* argument.

Use the **show access-lists ipv4 standby summary** command to display a summary of all standby IPv4 access lists.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

In this example, the contents of all IPv4 access lists are displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv4 standby summary

ACL Summary:
  Total ACLs configured: 4
  Total ACEs configured: 22
```

# show access-lists ipv6

To display the contents of current IPv6 access lists, use the **show access-lists ipv6** command in XR EXEC mode.

**show access-lists ipv6** [{*access-list-name* **hardware** {**ingress**|**egress**} **interface** *type interface-path-id* {**sequence** *number*|**location** *node-id*}|**summary** [*access-list-name*]|*access-list-name* [*sequence-number*] |**maximum** [**detail**] [**usage pfilter** { **location** *node-id* |**all**}]}]

## Syntax Description

| | |
|---|---|
| access-list-name | Name of a particular IPv6 access list. The name cannot contain a spaces or quotation marks, but can include numbers. |
| hardware | Identifies the access list as an access list for an interface. |
| ingress | Specifies an inbound interface. |
| egress | Specifies an outbound interface. |
| **sequence** *number* | (Optional) Sequence number of a particular IPv6 access list. Range is 1 to 2147483646. |
| interface | (Optional) Displays interface statistics. |
| type | Interface type. For more information, use the question mark (?) online help function. |
| interface-path-id | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | Location of a particular IPv4 access list. The *node-id* argument is entered in the *rack/slot/module* notation. |
| all | Displays the location of all the line cards. |
| summary | Displays a summary of all current IPv6 access lists. |
| sequence-number | (Optional) Sequence number of a particular IPv6 access list. Range is 1 to 2147483646. |
| maximum | Displays the current maximum number of configurable IPv6 access control lists (ACLs) and access control entries (ACEs). |
| detail | (Optional) Displays complete out-of-resource (OOR) details. |
| usage | (Optional) Displays the usage of the access list on a given line card. |
| pfilter | Displays the packet filtering usage for the specified line card. |

## Command Default

Displays all IPv6 access lists.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |
| Release 5.2.1 | The show command was updated to display ACL-based forwarding information. |

## Usage Guidelines

The **show access-lists ipv6** command is similar to the **show access-lists ipv4** command, except that it is IPv6 specific.

Use the **show access-lists ipv6** command to display the contents of all IPv6 access lists. To display the contents of a specific IPv6 access list, use the *name* argument. Use the *sequence-number* argument to specify the sequence number of the access list.

Use the **hardware , ingress** or **egress**, and **location** keywords to display the access list hardware contents and counters for all interfaces that use the specified access list in a given direction (ingress or egress). To display the contents of a specific access list entry, use the **sequence** *number* keyword and argument. The access group for an interface must be configured using the **ipv6 access-group** command for access list hardware counters to be enabled.

Use the **show access-lists ipv6 summary** command to display a summary of all current IPv6 access lists. To display a summary of a specific IPv6 access list, use the *name* argument.

Use the **show access-lists ipv6 maximum detail** command to display the OOR details for IPv6 access lists. OOR limits the number of ACLs and ACEs that can be configured in the system. When the limit is reached, configuration of new ACLs or ACEs is rejected.

Use the **show access-list ipv6 ipv4 usage** command to display a summary of all interfaces and access lists programmed on the specified line card.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

In the following example, the contents of all IPv6 access lists are displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv6

ipv6 access-list Internetfilter
  3 remark Block BGP traffic from a given host
  4 deny tcp host 6666:1:2:3::10 eq bgp host 7777:1:2:3::20 range 1300 1404 deny tcp host
171.69.2.88 255.255.0.0 any eq telnet
  20 permit ipv6 3333:1:2:3::/64 any
  25 permit ipv6 4444:1:2:3::/64 any
  30 permit ipv6 5555:1:2:3::/64 any
ipv6 access-list marketing
  10 permit ipv6 7777:1:2:3::/64 any (51 matches)
  20 permit ipv6 8888:1:2:3::/64 any (26 matches)
  30 permit ipv6 9999:1:2:3::/64 any (5 matches)
```

In the following example, the contents of an access list named Internetfilter is displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv6 Internetfilter
```

```
ipv6 access-list Internetfilter
  3 remark Block BGP traffic from a given host
  4 deny tcp host 6666:1:2:3::10 eq bgp host 7777:1:2:3::20 range 1300 1404 deny tcp host
171.69.2.88 255.255.0.0 any eq telnet
  20 permit ipv6 3333:1:2:3::/64 any
  25 permit ipv6 4444:1:2:3::/64 any
  30 permit ipv6 5555:1:2:3::/64 any
```

In the following example, the contents of an access list named acl_hw_1 is displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv6 acl_hw_1 hardware egress location 0/2/cp0

ipv6 access-list acl_hw_1
  10 permit icmp any any (251 hw matches)
  20 permit ipv6 3333:1:2:3::/64 any (29 hw matches)
  30 deny tcp any any (58 hw matches)
```

This table describes the significant fields shown in the display.

***Table 5: show access-lists ipv6 hardware Field Descriptions***

| Field | Description |
|---|---|
| hw matches | Number of hardware matches. |

In the following example, a summary of all IPv6 access lists is displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv6 summary

ACL Summary:
  Total ACLs configured: 3
  Total ACEs configured: 11
```

This table describes the significant fields shown in the display.

***Table 6: show access-lists ipv6 summary Field Descriptions***

| Field | Description |
|---|---|
| Total ACLs configured | Number of configured IPv6 ACLs. |
| Total ACEs configured | Number of configured IPV6 ACEs. |

In the following example, the OOR details of the IPv6 access lists are displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv6 maximum detail

Default max configurable acls :1000
Default max configurable aces :50000
Current configured acls       :1
Current configured aces       :2
Current max configurable acls :1000
Current max configurable aces :50000
Max configurable acls         :2000
Max configurable aces         :100000
```

This table describes the significant fields shown in the display.

*Table 7: show access-lists pv6 maximum detail Field Descriptions*

| Field | Description |
|---|---|
| Default max configurable acls | Default maximum number of configurable IPv6 ACLs allowed. |
| Default max configurable aces | Default maximum number of configurable IPv6 ACEs allowed. |
| Current configured acls | Number of configured IPv6 ACLs. |
| Current configured aces | Number of configured IPv6 ACEs. |
| Current max configurable acls | Configured maximum number of configurable IPv6 ACLs allowed. |
| Current max configurable aces | Configured maximum number of configurable IPv6 ACEs allowed. |
| Max configurable acls | Maximum number of configurable IPv6 ACLs allowed. |
| Max configurable aces | Maximum number of configurable IPv6 ACEs allowed. |

## Related Commands

| Command | Description |
|---|---|
| copy access-list ipv6, on page 11 | Copies an existing IPv6 access list. |
| deny (IPv6) , on page 22 | Sets the deny conditions for an IPv6 access list. |
| ipv6 access-list, on page 34 | Defines an IPv6 access list and enters IPv6 access list configuration mode. |
| permit (IPv6) , on page 49 | Set permit conditions for an IPv6 access list. |
| remark (IPv6) , on page 56 | Inserts a helpful remark about an IPv6 access list entry. |
| resequence access-list ipv6 , on page 60 | Changes the starting entry number of the first statement in an existing IPv4 access list, and the number by which subsequent statements are incremented. |

# show access-lists ipv6 standby

To display the contents of current IPv6 standby access lists, use the **show access-lists ipv6 standby** command in XR EXEC mode.

**show access-lists ipv6 standby** [*access-list-name*] [**summary**]

## Syntax Description

| | |
|---|---|
| access-list name | (Optional) Name of a particular IPv6 access list. The name cannot contain spaces or quotation marks, but can include numbers. |
| summary | (Optional) Displays a summary of all current IPv6 standby access lists. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced . |

## Usage Guidelines

Use the **show access-lists ipv6 standby** command to display the contents of current IPv6 standby access lists. To display the contents of a specific IPv6 access list, use the *name* argument.

Use the **show access-lists ipv6 standby summary** command to display a summary of all standby IPv6 access lists.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

In this example, the contents of all IPv6 standby access lists are displayed:

```
RP/0/RP0/CPU0:router# show access-lists ipv6 standby summary

ACL Summary:
  Total ACLs configured: 4
  Total ACEs configured: 22
```

This table describes the significant fields shown in the display.

***Table 8: show access-lists ipv6 standby summary Field Descriptions***

| Field | Description |
|---|---|
| Total ACLs configured | Number of configured standby IPv6 ACLs. |
| Total ACEs configured | Number of configured standby IPV6 ACEs. |

## Related Commands

| Command | Description |
|---|---|
| copy access-list ipv6, on page 11 | Copies an existing IPv6 access list. |
| ipv6 access-list, on page 34 | Defines an IPv6 access list and enters IPv6 access list configuration mode. |

# ARP Commands

This chapter describes the commands used to configure and monitor the Address Resolution Protocol (ARP ).

For detailed information about ARP concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# arp

To add a permanent entry in the Address Resolution Protocol (ARP) cache, use the **arp** command in XR Config mode. To remove an entry from the ARP cache, enter the **no** form of this command.

**arp** *ip-address hardware-address encapsulation-type* [**alias**]
**no arp** *ip-address hardware-address encapsulation-type* [**alias**]

## Syntax Description

| | |
|---|---|
| ip-address | IPv4 (network layer) address for which a permanent entry is added to the ARP cache. Enter the IPv4 address in a four-part dotted-decimal format that corresponds to the local data-link address (a 32-bit address). |
| hardware-address | Hardware (data link layer) address that the IPv4 address is linked to. Enter the local data-link address (a 48-bit address), such as 0800.0900.1834. |
| encapsulation-type | Encapsulation type. The encapsulation types are:<br>• arpa<br>• srp<br>• srpa<br>• srpb<br>For Ethernet interfaces, this is typically the arpa keyword. |
| alias | (Optional) Causes the software to respond to ARP requests as if it were the owner of both the specified IP address and hardware address, whether proxy ARP is enabled or not. |

## Command Default

No entries are permanently installed in the ARP cache.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The software uses ARP cache entries to translate 32-bit IP addresses into 48-bit hardware addresses.

Because most hosts support dynamic resolution, you generally need not specify static ARP cache entries.

Static entries are permanent entries that map a network layer address (IPv4 address) to a data-link layer address (MAC address). If the **alias** keyword is specified when creating the entry, the interface to which the entry is attached will act as if it is the owner of the specified addresses, that is, it will respond to ARP request packets for this network layer address with the data link layer address in the entry.

The software does not respond to any ARP requests received for the specified IP address unless proxy ARP is enabled on the interface on which the request is received. When proxy ARP is enabled, the software responds to ARP requests with its own local interface hardware address.

To remove all nonstatic entries from the ARP cache, enter the clear arp-cache, on page 84 in XR EXEC mode.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read, write |

## Examples

The following is an example of a static ARP entry for a typical Ethernet host:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# arp 192.168.7.19 0800.0900.1834 arpa
```

## Related Commands

| Command | Description |
|---|---|
| clear arp-cache, on page 84 | Deletes all dynamic entries from the ARP cache. |
| show arp, on page 88 | Displays the ARP cache. |

# arp learning

To enable the dynamic learning of ARP entries for a local subnet or all subnets, use the **arp learning** command.

To disable this command, use the **no** prefix or the **disable** option for this command.

**arp learning local**
**no arp learning local**
**arp learning disable**
**no arp learning disable**

## Syntax Description

| | |
|---|---|
| **local** | Enables the dynamic learning of ARP entries for local subnets.<br>When arp learning local is configured on an interface or sub-interface, it learns only the ARP entries from ARP packets on the same subnet. |
| **disable** | Disables the dynamic learning of all ARP entries. |

## Command Default

This command has no keywords or arguments.

## Command Modes

Sub-interface configuration mode

```
RP/0/RP0/CPU0:router(config)#interface GigabitEthernet 0/0/0/1
RP/0/RP0/CPU0:router(config-if)# ipv4 address 12.1.3.4 255.255.255.0
RP/0/RP0/CPU0:router(config-if)# arp learning local
RP/0/RP0/CPU0:router(config-if)# no shut
RP/0/RP0/CPU0:router(config-if)# commit
```

```
RP/0/RP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/1
RP/0/RP0/CPU0:router(config-if)# ipv4 address 12.1.3.4 255.255.255.0
RP/0/RP0/CPU0:router(config-if)# arp learning disable
RP/0/RP0/CPU0:router(config-if)# commit
```

# arp purge-delay

To delay purging Address Resolution Protocol (ARP) entries when an interface goes down, use the **arp purge-delay** command in interface configuration mode. To turn off the purge delay feature, use the **no** form of this command.

**arp purge-delay** *value*
**no arp purge-delay** *value*

## Syntax Description

| | |
|---|---|
| *v alue* | Sets the purge delay time in seconds. Range is 1 to 65535. |

## Command Default

Default value is off.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **arp purge-delay** command to delay purging ARP entries when an interface goes down. If the interface comes up within the delay time, then the ARP entries are restored to prevent packet loss with Equal Cost Multipath (ECMP) configured.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read, write |

## Examples

The following is an example of setting the purge delay to 50 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/7/0/0
RP/0/RP0/CPU0:router(config-if)# arp purge-delay 50
```

# arp timeout

To specify how long dynamic entries learned on an interface remain in the Address Resolution Protocol (ARP) cache, enter the **arp timeout** command in interface configuration mode. To remove the **arp timeout** command from the configuration file and restore the system to its default condition with respect to this command, enter the **no** form of this command.

**arp timeout** *seconds*
**no arp timeout** *seconds*

## Syntax Description

| | |
|---|---|
| seconds | Indicates the time, in seconds, for which an entry remains in the ARP cache. Range is 30 to 2144448000. |

## Command Default

Entries remain in the ARP cache for 14,400 seconds (4 hours).

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

This command is ignored when issued on interfaces that do not use ARP. Also, ARP entries that correspond to the local interface or that are statically configured by the user never time out.

The **arp timeout** command applies only to the interface that is entered. When the timeout is changed for an interface the change applies only to that interface.

The **show interfaces** command displays the ARP timeout value in hours:minutes:seconds, as follows:

```
ARP type: ARPA, ARP Timeout 04:00:00
```

## Task ID

| Task ID | Operations |
|---|---|
| cef | read, write |

## Examples

The following example shows how to set the ARP timeout to 3600 seconds to allow entries to time out more quickly than the default:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/7/0/0
RP/0/RP0/CPU0:router(config-if)# arp timeout 3600
```

## Related Commands

| Command | Description |
|---|---|
| clear arp-cache, on page 84 | Deletes all dynamic entries from the ARP cache. |
| show arp, on page 88 | Displays the ARP cache. |
| show interfaces | Displays statistics for all interfaces configured on the networking device.<br>For information on using the **show interfaces** command, see Cisco IOS XR software *Interface and Hardware Component Command Reference*. |

# clear arp-cache

To delete all dynamic entries from the Address Resolution Protocol (ARP) cache, clear the fast-switching cache, and clear the IP route cache; use the **clear arp-cache** command in XR EXEC mode.

**clear arp-cache** {**traffic** *type interface-path-id* **location** *node-id* | **location** *node-id*}

## Syntax Description

| | |
|---|---|
| traffic | Deletes statistics on the specified interface. |
| *t ype* | Interface type. For more information, use the question mark (?) online help function. |
| *interface- path-id* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the modular services card or line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on the interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | Clears the ARP entries for a specified location. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

When issued without keywords or arguments, the **clear arp-cache** command clears all entries in the ARP cache.

Configuration of the **clear arp-cache drop-adjacency** command on a particular location is not recommended. If the command is used on a bundle interface, then drop adjacencies may be deleted in one of the line cards and not on other line cards. This scenario can result in entry mismatch. You can use the **clear arp-cache**

**drop-adjacency** ***interface location*** **all** to remove drop adjacency that is learned for the interface on all the line cards.

## Task ID

| Task ID | Operations |
|---|---|
| cef | execute |

## Examples

The following example shows how to remove traffic statistic entries from the ARP cache that match the specified interface:

```
Router# clear arp-cache traffic HundredGigE 0/7/0/0 location 0/1/CPU0
```

The following example shows how to remove entries from the ARP cache that match the specified location:

```
Router# clear arp-cache location 0/1/CPU0
```

## Related Commands

| Command | Description |
|---|---|
| show arp, on page 88 | Displays the ARP cache. |

# local-proxy-arp

To enable local proxy Address Resolution Protocol (ARP) on an interface, enter the **local-proxy-arp** command in interface configuration mode. To disable local proxy ARP on the interface, enter the **no** form of this command.

**local-proxy-arp**
**no local-proxy-arp**

## Syntax Description

This command has no keywords or arguments.

## Command Default

Local proxy ARP is disabled on all interfaces.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

When local proxy ARP is enabled, the networking device responds to ARP requests that meet all the following conditions:

- The target IP address in the ARP request, the IP address of the ARP source, and the IP address of the interface on which the ARP request is received are on the same Layer 3 network.
- The next hop for the target IP address is through the same interface as the request is received.

Typically, local proxy ARP is used to resolve MAC addresses to IP addresses in the same Layer 3 network such as, private VLANs that are Layer 2-separated. Local proxy ARP supports all types of interfaces supported by ARP and unnumbered interfaces.

Using the **no** form of the command removes the specified command from the configuration file and restores the system to its default condition with respect to the command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read, write |

## Examples

The following example shows how to enable local proxy ARP on TenGigE interface 0/0/0/0:

```
RP/0/RP0/CPU0:router#(config)# interface TenGigE 0/0/0/0
RP/0/RP0/CPU0:router#(config-if)# local-proxy-arp
```

# proxy-arp

To enable proxy Address Resolution Protocol (ARP) on an interface, enter the **proxy-arp** command in interface configuration mode. To disable proxy ARP on the interface, enter the **no** form of this command.

**proxy-arp**
**no proxy-arp**

## Syntax Description

This command has no keywords or arguments.

## Command Default

Proxy ARP is disabled on all interfaces.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

When proxy ARP is disabled, the networking device responds to ARP requests received on an interface only if one of the following conditions is met:

- The target IP address in the ARP request is the same as the interface IP address on which the request is received.
- The target IP address in the ARP request has a statically configured ARP alias.

When proxy ARP is enabled, the networking device also responds to ARP requests that meet all of the following conditions:

- The target IP address is not on the same physical network (LAN) on which the request is received.
- The networking device has one or more routes to the target IP address.
- All of the routes to the target IP address go through interfaces other than the one on which the request is received.

Using the **no** form of the command removes the specified command from the configuration file and restores the system to its default condition with respect to the command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read, write |

## Examples

The following example shows how to enable proxy ARP on HundredGigE interface 0/7/0/0:

```
RP/0/RP0/CPU0:router#(config)# interface HundredGigE 0/7/0/0
RP/0/RP0/CPU0:router#(config-if)# proxy-arp
```

# show arp

To display the Address Resolution Protocol (ARP), enter the **show arp** command in XR EXEC mode.

**show arp [traffic]** [{*ip-address hardware-address* | *type interface-path-id* }] **location** *node-id*

## Command Default

The active is the default location.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

ARP establishes correspondences between network addresses (an IP address, for example) and Ethernet hardware addresses. A record of each correspondence is kept in a cache for a predetermined amount of time. As this time gets over, the records are refreshed after two unicast requests by ARP to the host IP address. If no response is received from the host, then the entry is cleared from the database.

For **show arp** *interface-type interface-instance* form, the **location** and *node-id* keyword and argument is mandatory for Bundle and VLAN-on-Bundle interfaces. These keywords and arguments indicate the location for which the cache entries for the bundle are to be displayed. For physical interfaces, specifying the **location** and *node-id* keyword and argument is optional since the interface can only exist on one node.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show arp** command with no location specified:

```
Router# show arp

-------------------------------------------------------------------------------
0/3/CPU0
-------------------------------------------------------------------------------

Address         Age        Hardware Addr   State      Type  Interface

.4.1.1       -          000c.cfe6.3336  Interface  ARPA HundredGigE0/3/1/3

.4.1.2       01:37:50   0000.c004.0102  Dynamic    ARPA HundredGigE0/3/1/3

.1.4.2         - 000c.cfe6.33b5  Interface  ARPA HundredGigE0/3/3/4

.1.0.2         - 000c.cfe6.33b1  Interface  ARPA HundredGigE0/3/3/0

.1.0.1         00:37:56 000a.8b08.857a  Dynamic    ARPA HundredGigE0/3/3/0

.1.4.1         01:37:51 000a.8b08.857e  Dynamic    ARPA HundredGigE0/3/3/4

.11.1.1     - 000c.cfe6.32fa  Interface  ARPA FastEthernet0/3/0/6

.1.5.2         - 000c.cfe6.33b6  Interface  ARPA FastEthernet0/3/3/5
```

```
.1.1.2         - 000c.cfe6.33b2  Interface  ARPA FastEthernet0/3/3/1

.1.1.1         01:37:51 000a.8b08.857b  Dynamic    ARPA FastEthernet0/3/3/1

.1.5.1         01:37:50 000a.8b08.857f  Dynamic    ARPA FastEthernet0/3/3/5

-------------------------------------------------------------------------------
0/2/CPU0
-------------------------------------------------------------------------------
Address         Age        Hardware Addr   State      Type  Interface

.6.9.1 01:11:55   0003.fe4c.0bff  Dynamic    ARPA MgmtEth0/2/CPU0/0

.6.25.6 01:09:29   000c.cfe6.2000  Dynamic    ARPA MgmtEth0/2/CPU0/0

.6.5.10 00:39:58   0009.7b49.0bff  Dynamic    ARPA MgmtEth0/2/CPU0/0
```

The following is sample output from the **show arp** command with the *interface-type interface-instance* argument:

```
Router# show arp MgmtEth 0/RP1/CPU0/0

Address         Age        Hardware Addr   State      Type  Interface
10.4.9.2        00:35:55   0030.7131.abfc  Dynamic    ARPA  MgmtEth0/RP1/CPU0/0
10.4.9.1        00:35:55   0000.0c07.ac24  Dynamic    ARPA  MgmtEth0/RP1/CPU0/0
10.4.9.99       00:49:12   0007.ebea.44d0  Dynamic    ARPA  MgmtEth0/RP1/CPU0/0
10.4.9.199      -          0001.c9eb.dffe  Interface  ARPA  MgmtEth0/RP1/CPU0/0
```

The following is sample output from the **show arp** command with the *hardware-address* designation:

```
Router# show arp 0005.5f1d.8100

Address Age Hardware Addr State Type Interface
172.16.7.2 - 0005.5f1d.8100 Interface ARPA HundredGigE2/0/1/2
```

The following is sample output from the **show arp** command with the **location** keyword and *node-id* argument:

```
Router# show arp location 0/2/CPU0

Address Age Hardware Addr State Type Interface
192.168.15.1 - 00dd.00ee.00ff Alias ARPA
192.168.13.1 - 00aa.00bb.00cc Static ARPA
172.16.7.1 00:35:49 0002.fc0e.9600 Dynamic ARPA HundredGigE2/0/1/2
172.16.7.2 - 0005.5f1d.8100 Interface ARPA HundredGigE2/0/1/2
```

# show arp idb

To display the ARP database statistics for an interface, use the **show arp idb** command in EXEC mode.

**show arp idb** *interface-name* **location** *node-id*

## Syntax Description

| | |
|---|---|
| *interface-name* | Name of the interface |
| *node-id* | Location of the interface. LC node for physical interfaces, RP or LC node for virtual interfaces |

## Command Default

There is no default location, location needs to be provided in the CLI.

## Command History

| Release | Modification |
|---|---|
| Release 3.3.0 | This command was introduced. |

## Usage Guidelines

The **show arp idb** command is useful to verify the IP addresses, Mac address, ARP configuration(s) applied on the interface and the entry statistics.

For **show arp idb** *interface-type interface-instance* form, the **location** *node-id* keyword and argument is mandatory for Bundle and VLAN-on-Bundle interfaces to indicate which location the cache entries for the bundle should be displayed.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show arp idb** command:

```
RP/0/0/CPU0:ios#show arp idb GigabitEthernet 0/0/0/0 location 0/0/CPU0
Mon Jan 30 10:32:15.387 IST
GigabitEthernet0/0/0/0 (0x00000060):
IDB Client: default
IPv4 address 1.1.1.1, Vrf ID 0x60000000
VRF Name default
Dynamic learning: Enable
Dynamic entry timeout: 14400 secs
Drop adjacency timeout: Disable
Purge delay: off
Cache limit: 128000
Incomplete glean count: 1
```

Complete glean count: 0

Complete protocol count: 0

Dropped glean count: 0

Dropped protocol count: 0

IPv4 caps added (state up)

MPLS caps not added

Interface not virtual, not client fwd ref,

Proxy arp not configured, not enabled

Local Proxy arp not configured

Packet IO layer is NetIO

Srg Role : DEFAULT

Idb Flag : 49292

IDB is Complete

IDB Flag Description:

[CAPS | COMPLETE | IPV4_CAPS_CREATED | SPIO_ATTACHED |

SPIO_SUPPORTED]

Idb Flag Ext : 0x0

Idb Oper Progress : NONE

Client Resync Time : Jan 30 10:07:10.736787

Total entries : 9

| Event Name | Time Stamp | S, M

| idb-create | Jan 30 10:07:10.784 | 1, 0

| idb-state-up | Jan 30 10:07:10.784 | 0, 0

| caps-state-update | Jan 30 10:07:10.784 | 0, 1

| address-update | Jan 30 10:07:10.784 | 0, 0

| idb-complete | Jan 30 10:07:10.784 | 0, 0

| idb-entry-create | Jan 30 10:07:10.784 | 0, 0

| idb-caps-add | Jan 30 10:07:10.784 | 0, 0

| idb-caps-add-cb | Jan 30 10:07:10.784 | 0, 0

| idb-last-garp-sent | Jan 30 10:07:11.808 | 0, 0

# show arp traffic

To display Address Resolution Protocol (ARP) traffic statistics, enter the **show arp traffic** command in XR EXEC mode.

**show arp traffic** [*type interface-path-id* ] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *type interface-path-id* | (Optional) Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the modular services card or line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/ RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on the interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Displays the ARP entry for a specific location. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

The active RP is the default location.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

ARP establishes correspondences between network addresses (an IP address, for example) and Ethernet hardware addresses. A record of each correspondence is kept in a cache for a predetermined amount of time and then discarded.

For **show arp traffic,** *interface-instance*, the **location** *node-id* keyword and argument is mandatory for Bundle and VLAN-on-Bundle interfaces. These keywords and arguments indicate the location for which the cache entries for the bundle are to be displayed. For physical interfaces, specifying the **location** *node-id* keyword and argument is optional because the interface can only exist on one node.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show arp traffic** command:

```
Router# show arp traffic

ARP statistics:
  Recv: 2691 requests, 91 replies
  Sent: 67 requests, 2 replies (0 proxy, 1 gratuitous)
  Resolve requests rcvd: 1
  Resolve requests dropped: 0
  Errors: 0 out of memory, 0 no buffers

ARP cache:
  Total ARP entries in cache: 5
  Dynamic: 3, Interface: 1, Standby: 0
  Alias: 0,   Static: 0, DHCP:0, DropAdj: 1

  IP Packet drop count for node 0/0/CPU0: 1
```

The following is sample output from the **show arp traffic** command with the **location** keyword and *node-id* argument:

## Related Commands

| Command | Description |
|---|---|
| clear arp-cache, on page 84 | Deletes all dynamic entries from the ARP cache. |
| show arp, on page 88 | Displays ARP statistics. |

# Cisco Express Forwarding Commands

This chapter describes the commands used to configure and monitor Cisco Express Forwarding (CEF) on .

For detailed information about CEF concepts, configuration tasks, and examples, see *Cisco IP Addresses and Services Configuration Guide*.

# cef adjacency route override rib

To enable the CEF prefer Routing Information Base (RIB) prefixes over Adjacency Information Base (AIB) prefixes in the Global configuration mode. To enable the CEF prefer AIB prefixes over RIB prefixes, use the **no** form of this command.

**cef adjacency route override rib**

**no cef adjacency route override rib**

## Syntax Description

| | |
|---|---|
| **route** | Enables adjacency route configuration<br>. |
| **override** | Sets override options for the adjacency routes. |
| **rib** | Sets options for adjacency routes to override the RIB routes. |

## Command Default

By default, CEF prefers RIB prefixes over AIB prefixes.

## Command Modes

Global configuration

## Command History

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

## Usage Guidelines

CEF may prefer the L2 adjacency for forwarding over the RIB (routing) entry under the following conditions:

- When there is no local ARP entry (yet).

  ARP learning may result in the router creating a forwarding entry.

- A forwarding entry of /32 (or /128 for IPv6) RIB routes are overridden when there is a covering connected or attached route.

  If an interface has a larger subnet, and you want to redirect a /32 out of that subnet of a different interface via a static route.

This can be seen in scenarios of EVPN and or HSRP, or in bridge domains with a BVI and multiple EFP's.

To deviate from the behavior of preferring a L2 adjacency for forwarding over a route entry, use the **cef adjacency route override rib** command.

## Task ID

| Task ID | Operation |
|---|---|
| cef | read, write |

### Example

The following example shows how to override the CEF adjacency route:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router# cef adjacency route override rib
```

# cef load-balancing algorithm adjust

To configure a rotate bit count value to adjust that is rotate the hash result so that it can vary from a next-hop router in a cascaded setup, use the **cef load-balancing algorithm adjust** command in global configuration mode. This command addresses traffic polarization issues in routers in a cascaded setup.

**cef load-balancing algorithm adjust** *value*

## Syntax Description

| | |
|---|---|
| *value* | This value is subject to a 'modulo' of 4 when applied on ASR 9000 Ethernet Line Card. For example, if the value configured is 10, the actual adjust value applied on ASR 9000 Ethernet Line Cards will be "10 mod 4" which is '2'. ASR 9000 Enhanced Ethernet Line Card will continue using the same adjust value as configured. Range is from 0 to 31.<br><br>**Note**: the hash shift command changes the hash result that is computed by the ingress linecard. This hash change affects both IPv4 and IPv6 for Equal Cost Multipath (ECMP) as well as the Bundle Member selection when used as either a routed (sub)-interface or as attachment circuit (AC) in L2VPN |

## Command Default

None

## Command Modes

Global configuration

## Command History

| Release | Modification |
|---|---|
| Release 4.2.3 | This command was introduced. |

## Usage Guidelines

This command has no effect on Layer 3 Multicast IP traffic.

## Task ID

| Task ID | Operation |
|---|---|
| config-services | read, write |

### Example

The following example shows how rotate bit count value to adjust the hash result:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# cef load-balancing algorithm adjust 2
```

# cef load-balancing fields

To select the hashing algorithm that is used for load balancing during forwarding, use the **cef load-balancing fields** command in XR Config mode. To undo a configuration and to default to the load balancing option of L3, use the **no** form of this command.

**cef load-balancing fields** {**L4**}**mplsentropy label**
**no cef load-balancing fields** {**L4**}

## Syntax Description

| | |
|---|---|
| **L3** | Specifies the Layer 3 load-balancing for the hash algorithm that is based on the following fields:<br>• Source IP address—Specifies the source IP address field in the IP packet header.<br>• Destination IP address—Specifies the destination IP address in the IP packet header.<br>• Router ID—Specifies the unique IP address that is assigned to the router.<br>Since L3 is configured as the default value, you do not need to use the **cef load-balancing fields** command unless you want to configure Layer 4. |

| | |
|---|---|
| L4 | Specifies the Layer 3 and Layer 4 load-balancing for the hash algorithm that is based on the following fields:<br>• Source IP address—Specifies the source IP address field in the IP packet header.<br>• Destination IP address—Specifies the destination IP address in the IP packet header.<br>• Source port—Specifies the value of the source port field in the TCP, UDP, or SCP packet header for Layer 4.<br>• Destination port—Specifies the value of the destination port field in the TCP, UDP, or SCP packet header for Layer 4.<br>• Router ID—Specifies the unique IP address that is assigned to the router.<br>• Protocol—Specifies the value of the protocol field as specified in the IP packet header for Layer 4.<br>• Slot Number:Rx UIDB Index—Specifies the slot number.<br>• Ingress interface—Specifies the interface that is received from the packet for Layer 4. |

## Command Default

When the router ID, source, and destination IP address fields are selected for load balancing, the default value is L3.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## 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.

You can undo only a Layer 4 configuration.

Per-flow load-balancing allows incoming data traffic on a router to be evenly distributed over multiple equal-cost connections. Per-flow load-balancing is the only load-balancing algorithm that is used to perform

Layer 2 and Layer 3 load-balancing decisions on IPv4, IPv6, and Multiprotocol Label Switching (MPLS) flows in the system.

The existing 3-tuple hash provides good-balancing for packet flows with different Layer 3 information (for example, source and destination IP addresses). However, this hash algorithm performs well for cases in which different packet flows, which are identified by Layer 4 content, contain the same Layer 3 packet information. For example, a network, which uses Port Address Translation (PAT) on one end of the network, distributes traffic to a content provider on the other end of the network that supports redundant access using the same IP address.

A new hash algorithm, which uses additional Layer 4 information from the Layer 3 packet, is needed to provide improved load-balancing support in the system. On the Cisco IOS XR software, the 7-tuple hash algorithm is implemented to provide improved load-balancing. The following inputs are processed:

- Layer 3 information
- Source IP address
- Destination IP address
- Protocol
- Layer 4 information
- Source port
- Destination port
- Cisco NCS router-related information
- Router ID
- Ingress interface bundle

**Note** The treatment is different for IPv4 and IPv6 fragmented packets that are fragmented on the router for various reasons. For example, the fragmented packets can originate at the router or they can arrive at the router with a size larger than the maximum transmission unit (MTU). Therefore, the 7-tuple load balancing is done on the whole packet, and fragmentation is done later so that all fragments can go on the same interface.

- Source IP address
- Destination IP address
- Router ID

**Note** This command has no effect on Layer 3 Multicast IP traffic.

**Task ID**

| Task ID | Operations |
|---|---|
| ipv4 | read, write |

**Examples**

The following example shows how to configure Layer 3 and Layer 4 load-balancing for the hash algorithm from the **cef load-balancing fields** command:

```
RP/0/RP0/CPU0:router# cef load balacing fields l3
```

## Related Commands

| Command | Description |
|---|---|
| show cef, on page 125 | Displays information about packets forwarded by Cisco Express Forwarding (CEF). |
| show cef summary, on page 133 | Displays summary information for the Cisco Express Forwarding (CEF) table. |
| show cef ipv4 exact-route, on page 143 | Displays an IPv4 Cisco Express Forwarding (CEF) exact route. |
| show cef ipv4 summary, on page 156 | Displays a summary of the IPv4 Cisco Express Forwarding (CEF) table |
| show cef ipv6 exact-route, on page 169 | Displays the path an IPv6 flow comprising a source and destination address would take. |
| show cef ipv6 summary, on page 181 | Displays a summary of the IPv6 Cisco Express Forwarding (CEF) table. |

# clear adjacency statistics

To clear adjacency packet and byte counter statistics, use the **clear adjacency statistics** command in XR EXEC mode.

**clear adjacency statistics** [{**ipv4** [**nexthop** *ipv4-address*] | **mpls** | **ipv6**}] [{*interface-type* *interface-instance* | **location** *node-id*}]

## Syntax Description

| | |
|---|---|
| ipv4 | (Optional) Clears only IPv4 adjacency packet and byte counter statistics. |
| **nexthop** *ipv4-address* | (Optional) Clears adjacency statistics that are destined to the specified IPv4 nexthop. |
| mpls | (Optional) Clears only MPLS adjacency statistics. |
| ipv6 | (Optional) Clears only IPv6 adjacency statistics. |
| interface-type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-instance* | (Optional) Either a physical interface instance or a virtual interface instance:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash mark between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( RP0 or RP1 ) and the module is CPU0. Example: interface MgmtEth0/ RP1 /CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Clears detailed adjacency statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **clear adjacency statistics** command is useful for troubleshooting network connection and forwarding problems.

If you do not specify any of the optional keywords, all adjacency statistics are cleared for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| cef | read, write |

## Related Commands

| Command | Description |
|---|---|
| show adjacency, on page 121 | Displays the IPv4 CEF adjacency table. |

# clear cef ipv4 drops

To clear Cisco Express Forwarding (CEF) IPv4 packet drop counters, use the **clear cef ipv4 drops**command in XR EXEC mode.

**clear cef ipv4 drops location** *node-id*

## Syntax Description

| | |
|---|---|
| **location** *node-id* | Clears IPv4 packet drop counters for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command will clear IPv4 CEF drop counters only for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| cef | read, write |

## Examples

The following example displays sample output for the IPv4 Cisco Express Forwarding (CEF) table packet drop counters, and clears IPv4 CEF drop counters for location 0/1/CPU0:

```
RP/0/RP0/CPU0:router# show cef ipv4 drops

CEF Drop Statistics
Node: 0/0/CPU0
  Unresolved drops     packets :              0
  Unsupported drops    packets :              0
  Null0 drops          packets :              0
  No route drops       packets :              0
  No Adjacency drops   packets :              0
  Checksum error drops packets :              0
  RPF drops            packets :              0
  RPF suppressed drops packets :              0
  RP destined drops    packets :              0
  Discard drops        packets :              0
  GRE lookup drops     packets :              0
  GRE processing drops packets :              0
Node: 0/1/CPU0
  Unresolved drops     packets :              0
  Unsupported drops    packets :              0
  Null0 drops          packets :              0
```

```
  No route drops       packets :               0
  No Adjacency drops   packets :               0
  Checksum error drops packets :               0
  RPF drops            packets :               0
  RPF suppressed drops packets :               0
  RP destined drops    packets :               0
  Discard drops        packets :               0
  GRE lookup drops     packets :               0
  GRE processing drops packets :               0
Node: 0/RP0/CPU0
  Unresolved drops     packets :               0
  Unsupported drops    packets :               0
  Null0 drops          packets :               0
  No route drops       packets :               0
  No Adjacency drops   packets :               0
  Checksum error drops packets :               0
  RPF drops            packets :               0
  RPF suppressed drops packets :               0
  RP destined drops    packets :               0
  Discard drops        packets :               0
  GRE lookup drops     packets :               0
  GRE processing drops packets :               0

RP/0/RP0/CPU0:router# clear cef ipv4 drops location 0/1/CPU0


   Node: 0/1/CPU0
   Clearing CEF Drop Statistics
```

## Related Commands

| Command | Description |
|---|---|
| show cef ipv4 drops, on page 141 | Displays IPv4 packet drop counters. |

# clear cef ipv4 exceptions

To clear IPv4 Cisco Express Forwarding (CEF) exception packet counters, use the **clear cef ipv4 exceptions** command in XR EXEC mode mode.

**clear cef ipv4 exceptions location** *node-id*

## Syntax Description

| | |
|---|---|
| **location** *node-id* | Clears IPv4 CEF exception packet counters for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command will clear IPv4 CEF exception packet counters for all nodes.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| cef | read, write |

## Examples

The following example displays sample output for the IPv4 Cisco Express Forwarding (CEF) exception packet counters, and clear s IPv4 CEF exception packets node 0/1/CPU0:

```
RP/0/RP0/CPU0:router# show cef ipv4 exceptions

CEF Exception Statistics
Node: 0/1/CPU0
  Slow encap  packets :              0
  Unsupported packets :              0
  Redirect    packets :              0
  Receive     packets :              0
  Broadcast   packets :              0
  IP options  packets :              0
  TTL expired packets :              0
  Fragmented  packets :              0
Node: 0/6/CPU0
  Slow encap  packets :              0
  Unsupported packets :              0
  Redirect    packets :              0
  Receive     packets :              0
  Broadcast   packets :              0
  IP options  packets :              0
```

```
  TTL expired packets :                0
  Fragmented  packets :                0
Node: 0/RP0/CPU0
  Slow encap  packets :                1
  Unsupported packets :                0
  Redirect    packets :                0
  Receive     packets :            71177
  Broadcast   packets :            23648
  IP options  packets :                0
  TTL expired packets :                0
  Fragmented  packets :                0
Node: 0/RP0/CPU0
  Slow encap  packets :                0
  Unsupported packets :                0
  Redirect    packets :                0
  Receive     packets :           167314
  Broadcast   packets :            22656
  IP options  packets :                0
  TTL expired packets :                0
  Fragmented  packets :                0

RP/0/RP0/CPU0:router# clear cef ipv4 exceptions location 0/1/CPU0

Node: 0/1/CPU0
Clearing CEF Exception Statistics
```

## Related Commands

| Command | Description |
|---|---|
| show cef ipv4 exceptions, on page 145 | Displays IPv4 CEF exception packet counters. |

# clear cef ipv4 interface bgp-policy-statistics

To clear Cisco Express Forwarding (CEF) IPv4 interface Border Gateway Protocol (BGP) policy statistics, use the **clear cef ipv4 interface bpg-policy-statistics** command in XR EXEC mode.

**clear cef ipv4 interface** *type interface-path-id* **bpg-policy-statistics**

## Syntax Description

| | |
|---|---|
| type | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

This command clears the Border Gateway Protocol (BGP) policy accounting counters for the specified interface.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| cef | read, write |

## Examples

The following example shows how to clear IPv4 CEF BGP policy statistics on a HundredGigE interface:

```
RP/0/RP0/CPU0:router# clear cef ipv4 interface HundredGigE 0/7/0/0 bgp-policy-statistics
```

## Related Commands

| Command | Description |
|---|---|
| show cef ipv4 interface bgp-policy-statistics, on page 150 | Displays IPv4 CEF BGP policy statistics. |

# clear cef ipv6 drops

To clear Cisco Express Forwarding (CEF) IPv6 packet drop counters, use the **clear cef ipv6 drop** command in XR EXEC mode.

**clear cef ipv6 drops location** *node-id*

## Syntax Description

| | |
|---|---|
| **location** *node-id* | Clears IPv6 packet drop counters for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command clears IPv6 CEF drop counters for all nodes.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| cef | read, write |

## Examples

The following example displays sample output for the IPv6 Cisco Express Forwarding (CEF) table packet drop counters, and clears IPv6 CEF drop counters for location 0/1/CPU0:

```
RP/0/RP0/CPU0:router# clear cef ipv6 drops

CEF Drop Statistics
Node: 0/1/CPU0
  Unresolved drops     packets :              0
  Unsupported drops    packets :              0
  Null0 drops          packets :              0
  No route drops       packets :              0
  No Adjacency drops   packets :              0
  Checksum error drops packets :              0
  RPF drops            packets :              0
  RPF suppressed drops packets :              0
  RP destined drops    packets :              0
Node: 0/6/CPU0
  Unresolved drops     packets :              0
  Unsupported drops    packets :              0
```

```
  Null0 drops          packets :              0
  No route drops       packets :              0
  No Adjacency drops   packets :              0
  Checksum error drops packets :              0
  RPF drops            packets :              0
  RPF suppressed drops packets :              0
  RP destined drops    packets :              0
Node: 0/RP0/CPU0
  Unresolved drops     packets :              0
  Unsupported drops    packets :              0
  Null0 drops          packets :              0
  No route drops       packets :              0
  No Adjacency drops   packets :              0
  Checksum error drops packets :              0
  RPF drops            packets :              0
  RPF suppressed drops packets :              0
  RP destined drops    packets :              0
Node: 0/RP0/CPU0
  Unresolved drops     packets :              0
  Unsupported drops    packets :              0
  Null0 drops          packets :              0
  No route drops       packets :              0
  No Adjacency drops   packets :              0
  Checksum error drops packets :              0
  RPF drops            packets :              0
  RPF suppressed drops packets :              0
  RP destined drops    packets :              0

RP/0/RP0/CPU0:router# clear cef ipv6 drop

Node: 0/1/CPU0
Clearing CEF Drop Statistics
```

## Related Commands

| Command | Description |
|---|---|
| show cef ipv6 drops, on page 167 | Displays IPv6 packet drop counters. |

# clear cef ipv6 exceptions

To clear IPv6 Cisco Express Forwarding (CEF) exception packet counters, use the **clear cef ipv6 exceptions** command in XR EXEC mode.

**clear cef ipv6 exceptions location** *node-id*

## Syntax Description

| | |
|---|---|
| **location** *node-id* | Clears IPv6 CEF exception packet counters for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command clears IPv6 CEF exception packet counters for all nodes.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| cef | read, write |

## Examples

The following example displays sample output for the IPv6 Cisco Express Forwarding (CEF) exception packet counters, and clears the IPv6 CEF exception packets for location:

```
RP/0/RP0/CPU0:router# show cef ipv6 exceptions

CEF Exception Statistics
Node: 0/1/CPU0
  Slow encap  packets :               0
  Unsupported packets :               0
  Redirect    packets :               0
  Receive     packets :               0
  Broadcast   packets :               0
  IP options  packets :               0
  TTL expired packets :               0
  Fragmented  packets :               0
Node: 0/6/CPU0
  Slow encap  packets :               0
  Unsupported packets :               0
  Redirect    packets :               0
  Receive     packets :               0
  Broadcast   packets :               0
  IP options  packets :               0
```

```
  TTL expired packets :               0
  Fragmented  packets :               0
Node: 0//CPU0
  Slow encap  packets :               0
  Unsupported packets :               0
  Redirect    packets :               0
  Receive     packets :               0
  Broadcast   packets :               0
  IP options  packets :               0
  TTL expired packets :               0
  Fragmented  packets :               0
Node: 0//CPU0
  Slow encap  packets :               0
  Unsupported packets :               0
  Redirect    packets :               0
  Receive     packets :               0
  Broadcast   packets :               0
  IP options  packets :               0
  TTL expired packets :               0
  Fragmented  packets :               0

RP/0/RP0/CPU0:router# clear cef ipv6 exceptions location 0/1/CPU0

Node: 0/1/CPU0
Clearing CEF Exception Statistics
```

## Related Commands

| Command | Description |
|---|---|
| show cef ipv6 exceptions, on page 171 | Displays IPv6 CEF exception packet counters. |

# clear cef ipv6 interface bgp-policy-statistics

To clear Cisco Express Forwarding (CEF) IPv6 interface Border Gateway Protocol (BGP) policy statistics, use the **clear cef ipv6 interface bpg-policy-statistics** command in XR EXEC mode.

**clear cef ipv6 interface** *type interface-path-id* **bpg-policy-statistics**

## Syntax Description

| | |
|---|---|
| type | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **clear cef ipv6 interface bgp-policy-statistics** command clears the Border Gateway Protocol (BGP) policy accounting counters for the specified interface.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write |
| cef | read, write |

## Examples

The following example shows how to clear IPv6 CEF BGP policy statistics:

```
RP/0/RP0/CPU0:router# clear cef ipv6 interface HundredGigE 0/7/0/0 bgp-policy-statistics
```

# ipv4 bgp policy propagation

To enable QoS Policy Propagation on BGP (QPPB) on an interface, use the **ipv4 bgp policy propagation** command in interface configuration mode. To disable QoS policy propagation on BGP, use the **no** form of this command.

**ipv4 bgp policy propagation input** {**ip-precedence** | **qos-group**} {**destination** | **source**}
**no ipv4 bgp policy propagation input** {**ip-precedence** | **qos-group**} {**destination** | **source**}

## Syntax Description

| | |
|---|---|
| input | Enables QPPB on the ingress IPv4 unicast interface. |
| ip-precedence | Specifies that the QoS policy is based on the IP precedence. |
| qos-group | Specifies that the QoS policy is based on the QoS group ID. |
| destination | Specifies that the IP precedence bit or QoS group ID from the destination address entry is used in the route table. |
| source | Specifies that the IP precedence bit or QoS group ID from the source address entry is used in the route table. |

## Command Default

The default is disabled.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

For the QPPB feature to work, you must enable BGP and CEF. In addition, the proper route-map configuration must be in place to specify the IP precedence or QoS group ID (for example, **set precedence** command).

If you specify both source and destination on the interface, the software looks up the source address in the routing table and classifies the packet based on the source address first; then the software looks up the destination address in the routing table and reclassifies it based on the destination address.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to enable QPPB on the HundredGigE interface:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/1/1/0
```

```
RP/0/RP0/CPU0:router(config-if)# ipv4 address 192.3.1.1 255.255.255.252
RP/0/RP0/CPU0:router(config-if)# ipv4 bgp policy propagation input ip-precedence destination
```

## Related Commands

| Command | Description |
|---|---|
| route-policy (BGP) | Defines a route policy. |
| show bgp policy | Displays information about BGP advertisements under a proposed policy. |
| show cef ipv4 interface bgp-policy-statistics, on page 150 | Displays IPv4 CEF BGP policy statistics. |
| show route | Displays the current routes for BGP in the RIB. |
| table-policy | Applies a routing policy to routes being installed into the routing table. For more information, see *Routing Command Reference for Cisco NCS 6000 Series Routers* |

# ipv4 verify unicast source reachable-via

To enable IPv4 unicast Reverse Path Forwarding (RPF) checking, use the **ipv4 verify unicast source reachable-via** command in an appropriate configuration mode. To disable unicast RPF, use the **no** form of this command.

**ipv4 verify unicast source reachable-via** {**any** | **rx**} [**allow-default**] [**allow-self-ping**]

## Syntax Description

| | |
|---|---|
| **any** | Enables loose unicast RPF checking. If loose unicast RPF is enabled, a packet is not forwarded unless its source prefix exists in the routing table. |
| **rx** | Enables strict unicast RPF checking. If strict unicast RPF is enabled, a packet is not forwarded unless its source prefix exists in the routing table and the output interface matches the interface on which the packet was received. |
| **allow-default** | (Optional) Enables the matching of default routes. This option applies to both loose and strict RPF. |
| **allow-self-ping** | (Optional) Enables the router to ping out an interface.This option applies to both loose and strict RPF. |

## Command Default

IPv4 unicast RPF is disabled.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **ipv4 verify unicast source reachable-via** interface command to mitigate problems caused by malformed or forged (spoofed) IP source addresses that pass through a router. Malformed or forged source addresses can indicate denial-of-service (DoS) attacks based on source IP address spoofing.

When strict unicast RPF is enabled on an interface, the router examines all packets received on that interface. The router checks to make sure that the source address appears in the routing table and matches the interface on which the packet was received.

When loose unicast RPF is enabled on an interface, the router examines all packets received on that interface. The router checks to make sure that the source address can be reached through any of the router interfaces.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example shows how to configure strict RPF on HundredGigE interface 0/7/0/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/7/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 verify unicast source reachable-via rx
```

This example shows how to configure loose RPF on HundredGigE interface 0/7/0/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/7/0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 verify unicast source reachable-via rx any
```

# rp mgmtethernet forwarding

To enable switching from the line card to the route processor Management Ethernet interfaces, use the **rp mgmtethernet forwarding** command in XR Config mode. To disable switching from the modular services card to the route processor Management Ethernet interfaces, use the **no** form of this command.

**rp mgmtethernet forwarding**
**no rp mgmtethernet forwarding**

## Syntax Description

This command has no keywords or arguments.

## Command Default

Switching is disabled.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The rp mgmtethernet forwarding command needs LC reload to take effect.

**Note** If enabled, the RP CPU is used to forward packets because the RP does not have a packet processing engine like the line cards.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read, write |

## Examples

The following example shows how to enable switching from the modular services card to the RP Management Ethernet interfaces:

```
RP/0/RP0/CPU0:router(config)# rp mgmtethernet forwarding
```

# show adjacency

To display Cisco Express Forwarding (CEF) adjacency table information, use the **show adjacency** command in XR EXEC mode.

**show adjacency** [{ **ipv4** [ **nexthop** *ipv4-address* ] | **mpls** | **ipv6** }] [ *interface type interface-instance* ] [**summary**] [**internal**] [**remote**] [**detail**] [ **location** *node-id* ]

## Syntax Description

| | |
|---|---|
| ipv4 | (Optional) Displays only IPv4 adjacencies. |
| **nexthop** *ipv4-address* | (Optional) Displays adjacencies that are destined to the specified IPv4 nexthop. |
| mpls | (Optional) Displays only MPLS adjacencies. |
| ipv6 | (Optional) Displays only IPv6 adjacencies. |
| interface-type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-instance* | Either a physical interface instance or a virtual interface instance:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash mark between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( RP0 or RP1 ) and the module is CPU0. Example: interface MgmtEth0/ RP1 /CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| summary | Displays summary of CEF IPv4, IPv6, MPLS adjacency counts for complete and incomplete entries in the adjacency table. |
| internal | Displays interfaces with internal HEX adjacencies and their hash values. |
| remote | (Optional) Displays only remote adjacencies. A remote adjacency is an internal adjacency used to forward packets between line cards. |
| detail | (Optional) Displays detailed adjacency information, including Layer 2 information. |

| | | |
|---|---|---|
| **location** | *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

This command is used to verify that an adjacency exists for a connected device, that the adjacency is valid, and that the MAC header rewrite string is correct.

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays the CEF adjacency table for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from **show adjacency** command with the **location** keyword specified:

```
RP/0/RP0/CPU0:router# show adjacency location 0/0/CPU0

Interface Address Version  Refcount  Protocol

/0/1/2(src mac only) 6 1 ipv4

/0/1/2 point to point 7 100004

/0/1/2                 (interface)                  3 1
```

The following is sample output from **show adjacency** command with the **ipv4** and **summary** keywords specified:

```
RP/0/RSP0/CPU0:ios#show adjacency ipv4 HundredGigE0/0/0/0 summary
Mon Feb 13 09:00:29.953 UTC
------------------------------------------------------------------------------
0/RSP0/CPU0
------------------------------------------------------------------------------

Adjacency table (version 1) has 1 adjacency:
------------------------------------------------------------------------------
0/0/CPU0
------------------------------------------------------------------------------

Adjacency table (version 4) has 4 adjacencies:
```

The following is sample output from **show adjacency** command with the **ipv4** and **detail** keywords specified:

```
RP/0/RSP0/CPU0:ios#show adjacency ipv4 HundredGigE0/0/0/0 detail
Mon Feb 13 09:05:22.086 UTC
------------------------------------------------------------------------------
```

```
0/RSP0/CPU0
------------------------------------------------------------------------------
Interface                  Address                 Version  Refcount Protocol
------------------------------------------------------------------------------
0/0/CPU0
------------------------------------------------------------------------------
Interface                  Address                 Version  Refcount Protocol
```

The following is sample output from **show adjacency** command with the **internal** and **location** keywords specified:

```
RP/0/RSP0/CPU0:ios#show adjacency internal location 0/RSP0/CPU0
Mon Feb 13 09:08:27.292 UTC
Interface                  Address                Entry     Protocol  HashIndex
Mg0/RSP0/CPU0/0            (interface)            0x7791d0a8          4447
```

The following is sample output from **show adjacency** command with the **internaldetail** and **location** keywords specified:

```
RP/0/RSP0/CPU0:ios#show adjacency internal detail location 0/RSP0/CPU0
Mon Feb 13 09:13:05.279 UTC

Mg0/RSP0/CPU0/0, (interface)
        Version: 1, references: 1, transient lock: 0
        MTU: 1500
        Adjacency pointer is: 0x7791d0a8
        Platform adjacency pointer is: 0x79d790a8
        Last updated: Feb 13 08:33:30.765
        Adjacency producer: dot1q (prod_id: 10)
        Flags: interface adjacency, incomplete adj,
                (Base-flag: 0x1, Entry-flag: 0x4, Status-flag: 0x0)
        Netio idb pointer not cached
        Cached interface type: 8
        Adjacency references:
                aib (JID 323, PID 6272), 1 reference
```

This table describes the significant fields shown in the display.

***Table 9: show adjacency Command Field Descriptions***

| Field | Description |
|---|---|
| Interface | Outgoing interface associated with the adjacency. |
| Address | Address can represent one of these addresses:<br>• Next hop IPv4 or IPv6 address<br>• Point-to-Point address<br>Information in parentheses indicates different types of adjacency. |
| Version | Version number of the adjacency. Updated whenever the adjacency is updated. |
| Refcount | Number of references to this adjacency. |
| Protocol | Protocol for which the adjacency is associated. |
| 0f000800 and 000c86f33d330800453a21c10800 | Layer 2 encapsulation string. |
| mtu | Value of the maximum transmission unit (MTU). |

| Field | Description |
|---|---|
| flags | Internal field. |
| packets | Number of packets going through the adjacency. |
| bytes | Number of bytes going through the adjacency. |

## Related Commands

| Command | Description |
|---|---|
| clear adjacency statistics, on page 104 | Clears the IPv4 CEF adjacency table. |

# show cef

To display information about packets forwarded by Cisco Express Forwarding (CEF), use the **show cef** command in XR EXEC mode.

**show cef** [*prefix* [*mask*]] [{**hardware** {**egress** | **ingress**} | **detail**}] [**location** {*node-id* | **all**}]

## Syntax Description

| | |
|---|---|
| prefix | (Optional) Longest matching CEF entry for the specified IPv4 destination prefix. |
| mask | (Optional) Exact CEF entry for the specified IPv4 prefix and mask. |
| hardware | (Optional) Displays detailed information about hardware. |
| egress | Displays information from the egress packet switch exchange (PSE) file. |
| ingress | Displays information from the ingress packet switch exchange (PSE) file. |
| detail | (Optional) Displays full details. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| all | (Optional) Displays all locations. |

## Command Default

When the prefix is not explicitly specified, this command displays all the IPv4 prefixes that are present in CEF. When not specified, the location defaults to the active Route Processor (RP) node.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output shows the load information flag from the **show cef** command for both **hardware** and **ingress** keywords:

```
RP/0/RP0/CPU0:router# show cef 101.1.3.0/24 hardware ingress location 0/3/CPU0
101.1.3.0/24, version 0, internal 0x40000001 (0x598491e8) [1], 0x0 (0x0),
(0x0)
 local adjacency 10.0.101.2
 Prefix Len 24, traffic index 0, precedence routine (0)
 BGP Attribute: id: 8, Local id: 6, Origin AS: 1003, Next Hop AS: 4

   via 10.0.101.2, 2 dependencies, recursive
```

```
    next hop 10.0.101.2 via 10.0.101.2/32


 Number of Mnodes:   2
 Mnode 0 HW Location: 0x00080404  HW Value
[ 0x0081a600 00000000 00000000 00000000 ]

 Leaf Mnode 1  HW Location: 0x040d3030
 Hardware Leaf:      PLU Leaf Value
[ 0x8000d800 028842c6 00000000 1fff2000 ]

 FCR  2 TLU Address 0x00210b19 TI 0 AS 6

VPN Label 1 0

 ************* IGP LoadInfo *****************
 Loadinfo HW Max Index  0
 Loadinfo SW Max Index  0
 PBTS Loadinfo Attached: No
 LI Path [ 0]  HFA Info: 0x10204028   FCR: 4
 ******************************************

-----------------------------------------------
 HW Rx Adjacency  0 Detail:
-----------------------------------------------
    Rx Adj HW Address   0x02040280  (ADJ)
    packets 0  bytes 0
    HFA Bits 0x80 gp 16 mtu 9248 (Fabric MTU) TAG length 0
    OI 0x409 (Tx uidb 0 PPindex 1033)
    OutputQ 0 Output-port 0x0 local-outputq 0x8000

[ 0x80181040 00002420 00000409 00008000 ]
[ 0x00000000 00000000 00000000 00000000 ]
[ 0x00000000 00000000 00000000 00000000 ]
```

# show cef bgp-attribute

To display Border Gateway Protocol (BGP) attributes for Cisco Express Forwarding (CEF), use the **show cef bgp-attribute** command in XR EXEC mode.

**show cef bgp-attribute** [**attribute-id** *index-id*] [**local-attribute-id** *index-id*] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **attribute-id** *index-id* | (Optional) Displays FIB attribute index. |
| **local-attribute-id** *index-id* | (Optional) Displays FIB local attribute index. |
| **location** *node-id* | (Optional) Displays BGP information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

The default location is active RP.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following example shows how to use the **show cef bgp-attribute** command:

```
RP/0/RP0/CPU0:router# show cef bgp-attribute

Total number of entries: 75742
BGP Attribute ID: 0x2058a, Local Attribute ID: 0x1
    Origin AS:    195, Next Hop AS:        195
BGP Attribute ID: 0x20583, Local Attribute ID: 0x2
    Origin AS:    22, Next Hop AS:        22
BGP Attribute ID: 0x20582, Local Attribute ID: 0x3
    Origin AS:    21, Next Hop AS:        21
BGP Attribute ID: 0x20585, Local Attribute ID: 0x4
    Origin AS:    28, Next Hop AS:        28
BGP Attribute ID: 0x20584, Local Attribute ID: 0x5
    Origin AS:    27, Next Hop AS:        27
BGP Attribute ID: 0x2057f, Local Attribute ID: 0x6
    Origin AS:    86, Next Hop AS:        86
BGP Attribute ID: 0x2058b, Local Attribute ID: 0x7
    Origin AS:    196, Next Hop AS:        196
BGP Attribute ID: 0x20589, Local Attribute ID: 0x8
    Origin AS:    194, Next Hop AS:        194
```

This table describes the significant fields shown in the display.

*Table 10: show cef bgp-attribute Command Field Descriptions*

| Field | Description |
|---|---|
| BGP Attribute ID | Displays the id assigned by BGP. |
| Local Attribute ID | Displays the id assigned by FIB. |
| Origin AS | Displays the origin AS of the prefix that carries this attribute id. |
| Next Hop AS | Displays the AS that contains the BGP nexthop for this prefix. |

## Related Commands

| Command | Description |
|---|---|
| show cef, on page 125 | Displays information about packets forwarded by Cisco Express Forwarding (CEF). |

# show cef external

To display Cisco Express Forwarding (CEF) external client dependency information, use the**show cef external** command in XR EXEC mode.

**show cef external** [**hardware** {**ingress** | **egress**}] [**prefix**] {*ifhandle tunnel-id client-name*} {**6vpe** | **6pe-ipvpn** | **eos0-ldi** | **ip-reachability**} [**detail**] [**location node-id**]

## Syntax Description

| | |
|---|---|
| hardware | (Optional) Displays hardware information. |
| ingress | (Optional) Displays hardware information programmed in ingress packet forwarding hardware. |
| egress | (Optional) Displays hardware information programmed in egress packet forwarding hardware. |
| prefix | (Optional) Displays external client information for a specific prefix. |
| ifhandle<br>**ifhandle** | Specifies interface handle. |
| tunnel-id<br>**tunnel-id** | Specifies the tunnel identifier. |
| client-name<br>**client-name** | Name of a particular client. The dependency information for the given client name is displayed. |
| 6vpe | Displays 6VPE (IPv6 VPN Provide Edge) dependency information. |
| 6vpe-ipvpn | Displays 6VPE over IP-VPN dependency information. |
| eos0-ldi | Displays Multiprotocol Label Switching (MPLS) end of stack 0 (EOS0) load balancing dependency information. |
| ip-reachability | Displays Internet Protocol (IP) reachability information. |
| detail | (Optional) Displays the dependency information in detail. |
| **location** *node-id* | (Optional) Displays external client dependency information for the specified node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

The following sample output is from the show cef external command:

```
RP/0/RP0/CPU0:router#show cef external hardware egress location 0/0/CPU0
Mon Dec 13 11:09:21.041 UTC

IPV4:
-----
Client Name       : l2fib_mgr (comp-id: 0x7e6d) (0x9f6f70fc)
Protocol          : ipv4
Prefix            : 3.3.3.3 (0x9f13d22c)
Gateway array     : 9e8fb058 (0x201500/1)
Loadinfo          : 9fbd41a8 (0x10181101/1)
Number of notifs  : 1
Interest type     : EOS0 LDI updates
Table Id          : 0xe0000000
Cookie Value      : 6c326669625f6d67720000000
State             : resolved, cached plat context
Via               : 16000/0
Added to pend list: Dec 13 11:08:37.920
    Load distribution: 0 (refcount 1)

    Hash  OK  Interface                 Address
    0     Y   HundredGigE0/0/0/9    10.0.9.2


Data identical on all NPs:

---- ECD LDI platform context data ----
 Flags: 0x21
 L2VPN LDI index: 0x1 (Search Key:0x100)
 Preferred path index: 0x5002dea0
 Cached L2FIB notification data:
    l2vpn_ldi_index: 0x1 (Search Key:0x100)
    recursion_level: 1 (RECURSION_NONE), num_paths: 1

       IGP Path info #0
       is_unresolved: 0
       Primary path: is_lag: 0, sfp_or_lagid: 1, ifhandle: 0x4000440
       Bkup path: is not valid
---- End of platform context data ----



RP/0/RP0/CPU0:router#show cef external hardware egress location 0/0/CPU0
Mon Dec 13 11:22:47.605 UTC

IPV4:
-----
Client Name       : l2fib_mgr (comp-id: 0x7e6d) (0x9f6f70fc)
Protocol          : ipv4
Prefix            : 100.100.100.2 (0x9f13d22c)
Gateway array     : 9e8fb058 (0x201500/1)
Loadinfo          : 9fbd41a8 (0x10181101/1)
Number of notifs  : 2
Interest type     : EOS0 LDI updates
Table Id          : 0xe0000000
Cookie Value      : 6c326669625f6d67720000000
State             : resolved, cached plat context
Via               : 16006/0
Added to pend list: Dec 13 11:21:23.037
```

```
    Load distribution: 0 (refcount 1)

    Hash  OK  Interface                 Address
    0     Y   recursive                 16006/0

Data identical on all NPs:

---- ECD LDI platform context data ----
 Flags: 0x21
 L2VPN LDI index: 0x2 (Search Key:0x200)
 Preferred path index: 0x5002dea8
 Cached L2FIB notification data:
    l2vpn_ldi_index: 0x2 (Search Key:0x200)
    recursion_level: 2 (RECURSION_ONE), num_paths: 1

       BGP Path info #0

         IGP Path info #0
         is_unresolved: 0
         Primary path: is_lag: 0, sfp_or_lagid: 1, ifhandle: 0x4000440
         Bkup path: is not valid
---- End of platform context data ----
```

## Related Commands

| Command | Description |
|---|---|
| show cef, on page 125 | Displays information about packets forwarded by Cisco Express Forwarding (CEF). |

# show cef recursive-nexthop

To display Cisco Express Forwarding (CEF) recursive next-hop information, use the**show cef recursive-nexthop** command in XR EXEC mode.

**show cef recursive-nexthop** [**hardware**] [**location node-id**]

## Syntax Description

| | |
|---|---|
| hardware | (Optional) Displays hardware information related to the recursive next hop. |
| **location** *node-id* | (Optional) Displays recursive next-hop information for the specified node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Related Commands

| Command | Description |
|---|---|
| show cef, on page 125 | Displays information about packets forwarded by Cisco Express Forwarding (CEF). |

# show cef summary

To display summary information for the Cisco Express Forwarding (CEF) table, use the **show cef summary** command in XR EXEC mode.

**show cef summary** [**location** {*node-id* | **all**}]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **all** | (Optional) Displays all locations. |

**Command Default**

The **show cef summary** command assumes the IPv4 CEF table and the active RP node as the location.

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operations |
|---|---|
| cef | read |

**Examples**

The following sample output is from the **show cef summary** command.

```
RP/0/RP0/CPU0:router# show cef summary location 0/1/CPU0

Router ID is 10.1.1.1

IP CEF with switching (Table Version 0) for node0_1_CPU0

  Load balancing: L3
  Tableid 0xe0000000,  Flags 0x301
   Refcount 318
  170 routes, 0 reresolve, 0 unresolved (0 old, 0 new), 12240 bytes
  183 load sharing elements, 57292 bytes, 184 references
  19 shared load sharing elements, 7036 bytes
  164 exclusive load sharing elements, 50256 bytes
  0 CEF route update drops, 10 revisions of existing leaves
  Resolution Timer: 15s
  0 prefixes modified in place
  0 deleted stale prefixes
  21 prefixes with label imposition, 60 prefixes with label information
Adjacency Table has 49 adjacencies
  25 incomplete adjacencies
```

This table describes the significant fields shown in the display.

***Table 11: show cef summary Command Field Descriptions***

| Field | Description |
|---|---|
| Load balancing | Current load-balancing mode. The default value is L3. |
| Table Version | Version of the CEF table. |
| tableid | Table identification number. |
| flags | Option value for the table |
| routes | Total number of routes. |
| reresolve | Total number of routes being reresolved. |
| unresolved (*x* old, *x* new) | Number of routes not yet resolved. |
| load sharing elements | Total number of internal load-sharing data structures. |
| bytes | Total memory used by internal load sharing data structures. |
| references | Total reference count of all internal load sharing data structures. |
| CEF resets | Number of CEF table resets. |
| revisions of existing leaves | Number of updates to existing prefixes. |
| Exponential (currently *x*s, peak *x*s) | Currently not used. |
| prefixes modified in place | Prefixes modified in place. |
| Adjacency Table has *x* adjacencies | Total number of adjacencies. |
| *x* incomplete adjacency | Total number of incomplete adjacencies. |

## Related Commands

| Command | Description |
|---|---|
| show cef, on page 125 | Displays information about packets forwarded by Cisco Express Forwarding (CEF). |

# show cef ipv4

To display the IPv4 Cisco Express Forwarding (CEF) table, use the **show cef ipv4** command in XR EXEC mode.

**show cef ipv4** [{*prefix* [*mask*] | *interface-type interface-instance*}] [**detail**] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| prefix | (Optional) Longest matching CEF entry for the specified IPv4 destination prefix. |
| mask | (Optional) Exact CEF entry for the specified IPv4 prefix and mask. |
| interface-type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-instance* | Either a physical interface instance or a virtual interface instance:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash mark between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| detail | (Optional) Displays full CEF entry information. |
| **location** *node-id* | (Optional) Displays the IPv4 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If the location is not specified, the command defaults to the active RP node.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays the CEF table on the node in which the command is issued. Otherwise, the command is effective on the node specified by the **location** *node-id* keyword and argument.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output is from the **show cef ipv4** command:

```
RP/0/RP0/CPU0:router/CPU0:router# show cef ipv4
Prefix              Next Hop            Interface
10.0.0.0/0          10.25.0.1           MgmtEth0/RP0/CPU0/0
10.0.0.0/32         broadcast
10.25.0.0/16        attached            MgmtEth0/RP0/CPU0/0
10.25.12.10/32      receive             MgmtEth0/RP0/CPU0/0
10.25.13.12/32      10.25.13.12         MgmtEth0/RP0/CPU0/0
10.25.16.11/32      10.25.16.11         MgmtEth0/RP0/CPU0/0
10.25.22.10/32      10.25.22.10         MgmtEth0/RP0/CPU0/0
10.25.26.10/32      10.25.26.10         MgmtEth0/RP0/CPU0/0
10.25.41.2/32       10.25.41.2          MgmtEth0/RP0/CPU0/0
10.25.41.5/32       10.25.41.5          MgmtEth0/RP0/CPU0/0
10.25.42.5/32       10.25.42.5          MgmtEth0/RP0/CPU0/0
10.25.44.15/32      10.25.44.15         MgmtEth0/RP0/CPU0/0
10.25.55.2/32       10.25.55.2          MgmtEth0/RP0/CPU0/0
10.25.255.255/32    10.25.255.255       MgmtEth0/RP0/CPU0/0
10.0.0.0/4          0.0.0.0
10.0.0.1/32         0.0.0.0
10.255.255.255/32   broadcast
```

This table describes the significant fields shown in the display.

***Table 12: show cef ipv4 Command Field Descriptions***

| Field | Description |
|---|---|
| Prefix | Prefix in the IPv4 CEF table. |
| Next Hop | Next hop of the prefix. |
| Interface | Interface associated with the prefix. |

# show cef ipv4 adjacency

To display Cisco Express Forwarding (CEF) IPv4 adjacency status and configuration information, use the **show cef ipv4 adjacency** command in XR EXEC mode.

**show cef ipv4 adjacency** [*interface-type interface-path-id*] [**location** *node-id*] [**detail**] [**discard**] [**glean**] [**null**] [**punt**] [**remote**] [**protected**]

## Syntax Description

| | |
|---|---|
| interface-type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface- path-id* | (Optional) Either a physical interface instance or a virtual interface instance:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash mark between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| detail | (Optional) Displays the detailed adjacency information. |
| discard | (Optional) Filters out and displays only the discarded adjacency information. |
| glean | (Optional) Filters out and displays only the glean adjacency information. |
| null | (Optional) Filters out and displays only the adjacency information. |
| punt | (Optional) Filters out and displays only the punt adjacency information. |
| remote | (Optional) Filters out and displays only the remote adjacency information. |
| protected | (Optional) Filters out and displays only the IP-Fast Reroute (FRR) protected adjacency information. |

**Command Default**

No default behavior or values

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

If you do not specify a node with the **location** keyword and *node-id* argument, the **show cef ipv4 adjacency** command displays the CEF adjacency table for the node on which the command is issued.

**Task ID**

| Task ID | Operations |
|---|---|
| cef | read |

**Examples**

The following sample output is from **show cef ipv4 adjacency** command :

```
RP/0/RP0/CPU0:router:# show cef ipv4 adjacency MgmtEth 0/RP0/CPU0/0
Display protocol is ipv4
Interface    Address                                        Type    Refcount
Mg0/RP0/CPU0/0Prefix: 10.25.0.3/32                           local   2
             Adjacency: PT:0x782a2900 12.25.0.3/32
             Interface: Mg0/RP0/CPU0/0
             MAC: 00.d0.02.75.ab.fd.00.11.93.ef.e3.50.08.00
             Interface Type: 0x8, Base Flags: 0x1
             Dependent adj type: remote
             Dependent adj intf: Mg0/RP0/CPU0/0
Mg0/

/CPU0/0Prefix: 10.24.0.32/32                               remote  6
             Adjacency: PT:0x782a2b58
             Interface: Mg0/RP0/CPU0/0
             MAC: 28.4e.4f.4e.45.29
             Interface Type: 0x8, Base Flags: 0x0
```

This table describes the significant fields shown in the display.

***Table 13: show cef ipv4 adjacency Command Field Descriptions***

| Field | Description |
|---|---|
| Interface | Interface associated with the prefix. |
| Address | Prefix address information. |
| Type | Type of adjacency, can be either local or remote. |
| Refcount | Number of times the adjacency is referenced by other routers. |

# show cef ipv4 adjacency hardware

To display Cisco Express Forwarding (CEF) IPv4 adjacency hardware status and configuration information, use the **show cef ipv4 adjacency hardware** command in XR EXEC mode.

**show cef ipv4 adjacency hardware** {**egress** | **ingress**} [{**detail** | **discard** | **drop** | **glean** | **location** *node-id* | **null** | **punt** | **protected** | **remote**}]

## Syntax Description

| | |
|---|---|
| egress | Displays information from the egress packet switch exchange (PSE) file. |
| ingress | Displays information from the ingress packet switch exchange (PSE) file. |
| detail | (Optional) Displays full details. |
| discard | (Optional) Displays the discard adjacency information. |
| drop | (Optional) Displays the drop adjacency information. |
| glean | (Optional) Displays the glean adjacency information. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| null | (Optional) Displays the null adjacency information. |
| punt | (Optional) Displays the punt adjacency information. |
| protected | (Optional) Filters out and displays only the IP-Fast Reroute (FRR) protected adjacency information. |
| remote | (Optional) Displays the remote adjacency information. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output shows the load information flag from the **show cef ipv4 adjacency hardware** command for the **egress** keyword:

```
RP/0/RP0/CPU0:router# show cef ipv4 adjacency hardware egress detail location 0/2/CPU0

Display protocol is ipv4
Interface    Address                                          Type    Refcount

tt0          Prefix: 0.0.0.0/32                               local   5
             no next-hop adj
             Interface: NULLIFHNDL
             Mac-length is 0
             tunnel interface
             Interface Type: 0x24, Base Flags: 0x2001
             Dependent adj type: remote
             Dependent adj intf: tt0

  TE Flags        : 0x41
  TLU3(temp)  : 0x200b801
  [HW: 0x00000001 0x20020000 0x08000000 0x00080000]
    type          : FWD
    num. entries  : 1
    uidb index    : 2
    num. labels   : 0
    label         : 0
    encapsulation : unknown (0x8000000)
    next ptr      : 0x800
  TLU4            : 0x3000800
    Entry[0]
    [HW: 0x00000080 0x0013c48f 0x880b05ea 0x00580000]
      label         : 0
      num. labels   : 0
      local         : 1
      mtu           : 1514
      default sharq : 11
      member link   : 0

Te0/2/0/1                                                     special 2
             Interface: Te0/2/0/1 Type:  glean
             Interface Type: 0x1e, Base Flags: 0x4400
             Dependent adj type: remote
             Dependent adj intf: Te0/2/0/1
TLU 3 Unavailable
```

This table describes the significant fields shown in the display.

***Table 14: show cef ipv4 adjacency hardware Command Field Descriptions***

| Field | Description |
|---|---|
| Interface | Interface associated with the prefix. |
| Address | Prefix address information. |
| Type | Type of adjacency, can be either local or remote. |
| Refcount | Number of times the adjacency is referenced by other routers. |

# show cef ipv4 drops

To display IPv4 Cisco Express Forwarding (CEF) table packet drop counters, use the **show cef ipv4 drops** command in XR EXEC mode.

**show cef ipv4 drops** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays IPv4 CEF table packet drop counters for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

No default behavior or values

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

A packet might be dropped from the IPv4 CEF table because of unresolved CEF entries, unsupported features, absence of route information, absence of adjacency information, or an IP checksum error.

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays IPv4 CEF packet drop counters for all nodes.

**Task ID**

| Task ID | Operations |
|---|---|
| cef | read |

**Examples**

The following is sample output from the **show cef ipv4 drops** for location command:

```
RP/0/RP0/CPU0:router# show cef ipv4 drops

CEF Drop Statistics
Node: 0/0/CPU0
  Unresolved drops     packets :              0
  Unsupported drops    packets :              0
  Null0 drops          packets :              0
  No route drops       packets :              0
  No Adjacency drops   packets :              0
  Checksum error drops packets :              0
  RPF drops            packets :              0
  RPF suppressed drops packets :              0
  RP destined drops    packets :              0
```

***Table 15: show cef ipv4 drop Command Field Descriptions***

| Field | Description |
|---|---|
| Unresolved drops | Drops due to unresolved routes. |

| Field | Description |
|---|---|
| Unsupported drops | Drops due to an unsupported feature. |
| Null0 drops | Drops to the Null0 interface. |
| No route drops | Number of packets dropped because there were no routes to the destination. |
| No Adjacency drops | Number of packets dropped because there were no adjacencies established. |
| Checksum error drops | Drops due to IPv4 checksum error. |
| RPF drops | Drops due to IPv4 unicast RPF[1]. |
| RPF suppressed drops | Drops suppressed due to IPv4 unicast RPF. |
| RP destined drops | Drops destined for the router. |

[1] RPF = Reverse Path Forwarding

## Related Commands

| Command | Description |
|---|---|
| clear cef ipv4 drops, on page 106 | Clears IPv4 CEF packet drop counters. |

# show cef ipv4 exact-route

To display an IPv4 Cisco Express Forwarding (CEF) exact route, use the **show cef ipv4 exact-route** command in XR EXEC mode.

**show cef ipv4 exact-route** {*source-address destination-address*} [**protocol** *protocol-name* **source-port** *source-port* **destination-port** *destination-port* **ingress-interface** *typeinterface-path-id*] [**policy-class** *policy-class-value*] [**detail | location** *node-id*]

## Syntax Description

| | |
|---|---|
| *source-address* | The IPv4 source address in x.x.x.x format. |
| destination-address | The IPv4 destination address in x.x.x.x format. |
| **protocol** *protocol name* | (Optional) Displays the specified protocol for the route. |
| **source-port** *source-port* | (Optional) Sets the UDP source port. The range is from 0 to 65535. |
| **destination-port** *destination-port* | (Optional) Sets the UDP destination port. The range is from 0 to 65535. |
| ingress-interface | (Optional) Sets the ingress interface. |
| type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **policy-class** *value* | (Optional) Displays the class for the policy-based tunnel selection. The range for the tunnel policy class value is from 1 to 7. |
| detail | (Optional) Displays full CEF entry information. |
| **location** *node-id* | (Optional) Displays the IPv4 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If the Layer 4 information is enabled, the source-port, destination-port, protocol, and ingress-interface fields are required. Otherwise, the output of the **show cef ipv4 exact-route** command is not correct.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output is from the **show cef ipv4 exact-route** command:

```
RP/0/RP0/CPU0:router# show cef ipv4 exact-route 10.1.1.1 10.1.1.2 detail

0.0.0.0/0, version 432, proxy default, internal 0x2000201[1]
 Prefix Len 0, traffic index 0, precedence routine (0)
   via MgmtEth0/RP1/CPU0/0
```

This table describes the significant fields shown in the display.

***Table 16: show cef ipv4 exact-route Command Field Descriptions***

| Field | Description |
|---|---|
| Prefix | Prefix in the IPv4 CEF table . |
| Next Hop | Next hop of the prefix |
| Interface | Interface associated with the prefix |

## Related Commands

| Command | Description |
|---|---|
| show mpls forwarding exact-route | Displays the path an MPLS flow that comprises a source and destination address would take. |

# show cef ipv4 exceptions

To display IPv4 Cisco Express Forwarding (CEF) exception packet counters, use the **show cef ipv4 exceptions** command in XR EXEC mode.

**show cef ipv4 exceptions** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays CEF exception packet counters for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

CEF exception packets are those packets that have been sent from the hardware to the software because they require additional handling. The types of IPv4 CEF exception packets are displayed in the command's output and are defined.

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays IPv4 CEF exception packet counters on all nodes.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show cef ipv4 exceptions** command:

```
RP/0/RP0/CPU0:router# show cef ipv4 exceptions

CEF Exception Statistics
Node: 0/0/CPU0
  Slow encap  packets :              0
  Redirect    packets :              0
  Receive     packets :         306404
  Broadcast   packets :              0
  IP options  packets :              0
  TTL expired packets :              0
  Fragmented  packets :              0
Node: 0/1/CPU0
  Slow encap  packets :              0
  Redirect    packets :              0
  Receive     packets :              0
  Broadcast   packets :              0
  IP options  packets :              0
  TTL expired packets :              0
  Fragmented  packets :              0
Node: 0/2/CPU0
```

```
  Slow encap  packets :               0
  Redirect    packets :               0 Receive     packets :               0
  Broadcast   packets :               0
  IP options  packets :               0
  TTL expired packets :             314
  Fragmented  packets :               0
Node: 0/3/CPU0
  Slow encap  packets :               0
  Redirect    packets :               0
  Receive     packets :               0
  Broadcast   packets :               0
  IP options  packets :               0
  TTL expired packets :               0
  Fragmented  packets :               0
```

This table describes the significant fields shown in the display.

***Table 17: show cef ipv4 exceptions Command Field Descriptions***

| Field | Description |
|---|---|
| Slow encap | Number of packets requiring special processing during encapsulation. |
| Redirect | Number of ICMP[2] redirect messages sent. |
| Receive | Number of packets destined to the router. |
| Broadcast | Number of broadcasts received. |
| IP options | Number of IP option packets. |
| TTL expired | Number of packets with expired TTLs[3]. |
| Fragmented | Number of packets that have been fragmented. |

[2] ICMP = internet control message protocol
[3] TTL = time to live

## Related Commands

| Command | Description |
|---|---|
| clear cef ipv4 exceptions, on page 108 | Clears IPv4 CEF exception packet counters. |

# show cef ipv4 hardware

To display Cisco Express Forwarding (CEF) IPv4 hardware status and configuration information, use the **show cef ipv4 hardware** command in XR EXEC mode.

**show cef ipv4 hardware** {**egress** | **ingress**}[{**detail** | **location** *node-id*}]

## Syntax Description

| | |
|---|---|
| egress | Displays information from the egress packet switch exchange (PSE) file. |
| ingress | Displays information from the ingress packet switch exchange (PSE) file. |
| detail | (Optional) Displays full details. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

# show cef ipv4 interface

To display IPv4 Cisco Express Forwarding (CEF)-related information for an interface, use the **show cef ipv4 interface** command in XR EXEC mode.

**show cef ipv4 interface** *type interface-path-id* [**detail**] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| type | Interface type. For more information, use the question mark (?) online help function. |
| *in terface-path-id* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the modular services card or line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/ RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| detail | (Optional) Displays detailed CEF information for all the interfaces on the node in which the command is issued. |
| **location** *node-id* | (Optional) Displays IPv4 CEF-related information for an interface. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show cef ipv4 interface** command:

```
RP/0/RP0/CPU0:router# show cef ipv4 interface MgmtEth 0/RP0/CPU0/0

MgmtEth0/0/CPU0/0 is up (if_handle 0x01000100)
  Forwarding is enabled
  ICMP redirects are never sent
  IP MTU 1500, TableId 0xe0000000
  Reference count 2
```

This table describes the significant fields shown in the display.

***Table 18: show cef ipv4 interface Command Field Descriptions***

| Field | Description |
|---|---|
| MgmtEth 0/RP0/CPU0/0 is up | Status of the interface. |
| if_handle | Internal interface handle. |
| Forwarding is enabled | Indicates that Cisco Express Forwarding (CEF) is enabled. |
| ICMP redirects are always sent or never sent | Indicates whether ICMP[4] redirect messages should be sent. By default, ICMP redirect messages are always sent. |
| IP MTU | Value of the IPv4 MTU[5] size set on the interface. |
| Reference count | Internal reference counter. |

[4] ICMP = internet control message protocol
[5] MTU = maximum transmission unit

# show cef ipv4 interface bgp-policy-statistics

To display IPv4 Cisco Express Forwarding (CEF)-related Border Gateway Protocol (BGP) policy statistics information for an interface, use the **show cef ipv4 interface bgp-policy-statistics** command in XR EXEC mode .

**show cef ipv4 interface** *type interface-path-id* **bgp-policy-statistics** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| type | Interface type. For more information, use the question mark (?) online help function. |
| interface-path-id | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Displays IPv4 CEF-related information for an interface. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

This command displays all the configured BGP policy counters for the specified interface.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show cef ipv4 interface bgp-policy-statistics** command:

```
RP/0/RP0/CPU0:router# show cef ipv4 interface HundredGigE 0/7/0/0 bgp-policy-statistics

HundredGigE 0/7/0/0 is up
Input BGP policy accounting on src IP address enabled
buckets packets  bytes
0       184054   10157753
6       65688590 4204069760
7       65688590 4204069760
8       65688654 4204073856
9       65688656 4204073984
10      65688655 4204073920
30      32844290 1510837340
31      32844291 1510837386
```

```
32      32844294 1510837524
33      32844296 1510837616
34      32844298 1510837708
35      32844302 1510837892
36      32844302 1510837892
37      32844303 1510837938
38      32844305 1510838030
39      32844307 1510838122
Output BGP policy accounting on dst IP address enabled
buckets packets bytes
0       754     43878
Output BGP policy accounting on src IP address enabled
buckets packets bytes
0       857     51706
```

This table describes the significant fields shown in the display.

***Table 19: show cef ipv4 interface bgp-policy-statistics Command Field Descriptions***

| Field | Description |
|---|---|
| HundredGigE 0/2/0/4 is up | Status of the interface. |
| Input BGP policy accounting on src IP address enabled | Enabled BGP policy accounting features. |
| buckets | Traffic index. |
| packets | Number of packets counted in the bucket. |
| bytes | Number of bytes counted in the bucket. |

# show cef ipv4 non-recursive

To display the IPv4 nonrecursive prefix entries in the IPv4 Cisco Express Forwarding (CEF) table, use the **show cef ipv4 non-recursive** command in XR EXEC mode.

**show cef ipv4 non-recursive** [**detail**] [**hardware** {**egress** | **ingress**}] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| detail | (Optional) Displays detailed information about nonrecursive prefix entries in the IPv4 CEF table. |
| hardware | (Optional) Displays detailed information about hardware. |
| egress | (Optional) Displays egress packet switch exchange (PSE). |
| ingress | (Optional) Displays ingress packet switch exchange (PSE). |
| interface-type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-instance* | (Optional) Either a physical interface instance or a virtual interface instance:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash mark between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Displays the IPv4 nonrecursive prefix entries in the IPv4 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, the output displays the IPv4 CEF nonrecursive routes for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show cef ipv4 non-recursive** command:

```
RP/0/RP0/CPU0:router#router# show cef ipv4 non-recursive

Prefix              Next Hop            Interface
0.0.0.0/0           1012.8.0.1
0.0.0.0/32          broadcast
10.8.0.0/16         attached            MgmtEth0/0/CPU0/0
10.8.0.0/32         broadcast           MgmtEth0/0/CPU0/0
10.8.0.1/32         12.8.0.1            MgmtEth0/0/CPU0/0
10.8.0.2/32         12.8.0.2            MgmtEth0/0/CPU0/0
10.8.0.3/32         12.8.0.3            MgmtEth0/0/CPU0/0
10.8.16.10/32       12.8.16.10          MgmtEth0/0/CPU0/0
10.8.16.30/32       12.8.16.30          MgmtEth0/0/CPU0/0
10.8.16.40/32       12.8.16.40          MgmtEth0/0/CPU0/0
10.8.28.8/32        12.8.28.8           MgmtEth0/0/CPU0/0
10.8.28.101/32      12.8.28.101         MgmtEth0/0/CPU0/0
10.8.28.103/32      12.8.28.103         MgmtEth0/0/CPU0/0
10.8.28.104/32      12.8.28.104         MgmtEth0/0/CPU0/0
10.8.28.106/32      receive             MgmtEth0/0/CPU0/0
10.8.29.113/32      12.8.29.113         MgmtEth0/0/CPU0/0
10.8.29.118/32      12.8.29.118         MgmtEth0/0/CPU0/0
10.8.29.140/32      12.8.29.140         MgmtEth0/0/CPU0/0
10.8.33.101/32      12.8.33.101         MgmtEth0/0/CPU0/0
10.8.33.103/32      12.8.33.103         MgmtEth0/0/CPU0/0
10.8.33.105/32      12.8.33.105         MgmtEth0/0/CPU0/0
10.8.33.110/32      12.8.33.110         MgmtEth0/0/CPU0/0
10.8.57.1/32        12.8.57.1           MgmtEth0/0/CPU0/0
10.8.255.255/32     broadcast           MgmtEth0/0/CPU0/0
10.29.31.2/32       12.29.31.2          MgmtEth0/0/CPU0/0
10.255.0.0/16            attached        MgmtEth0/0/CPU0/0
10.255.254.254/32   10223.255.254.254     MgmtEth0/0/CPU0/0
10.0.0.0/4            0.0.0.0
10.0.0.0/24         receive
255.255.255.255/32   broadcast
```

This table describes the significant fields shown in the display.

***Table 20: show cef ipv4 non-recursive Command Field Descriptions***

| Field | Description |
|---|---|
| Prefix | Nonrecursive prefixes detected on the node. |
| Next Hop | Routing next hop. |
| Interface | Interface associated with the nonrecursive prefix. |

# show cef ipv4 resource

To display the IPv4 nonrecursive prefix entries in the IPv4 Cisco Express Forwarding (CEF) table, use the **show cef ipv4 resource** command in XR EXEC mode.

**show cef ipv4 resource** [**detail**] [**hardware** {**egress** | **ingress**}] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| detail | (Optional) Displays detailed information resources listed in the IPv4 CEF table. |
| hardware | (Optional) Displays detailed information about hardware. |
| egress | (Optional) Displays egress packet switch exchange (PSE). |
| ingress | (Optional) Displays ingress packet switch exchange (PSE). |
| **location** *node-id* | (Optional) Displays the IPv4 resource entries in the IPv4 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, the output displays the IPv4 CEF nonrecursive routes for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show cef ipv4 resource** command:

```
RP/0/RP0/CPU0:router# show cef ipv4 resource detail

CEF resource availability summary state: GREEN
  ipv4 shared memory resource:
        CurrMode GREEN, CurrUtil 0%
        CurrAvail 1874526208 bytes, MaxAvail 1875693568 bytes
  ipv6 shared memory resource:
        CurrMode GREEN, CurrUtil 0%
        CurrAvail 1874591744 bytes, MaxAvail 1875365888 bytes
  mpls shared memory resource:
        CurrMode GREEN, CurrUtil 0%
        CurrAvail 1874407424 bytes, MaxAvail 1875038208 bytes
  common shared memory resource:
        CurrMode GREEN, CurrUtil 0%
        CurrAvail 1873215488 bytes, MaxAvail 1874972672 bytes
  TABLE hardware resource: GREEN
```

```
LEAF hardware resource: GREEN
LOADINFO hardware resource: GREEN
NHINFO hardware resource: GREEN
LABEL_INFO hardware resource: GREEN
IDB hardware resource: GREEN
FRR_NHINFO hardware resource: GREEN
LDSH_ARRAY hardware resource: GREEN
RSRC_MON hardware resource: GREEN
```

# show cef ipv4 summary

To display a summary of the IPv4 Cisco Express Forwarding (CEF) table, use the **show cef ipv4 summary** command in XR EXEC mode.

**show cef ipv4 summary** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays a summary of the IPv4 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

No default behavior or values

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays a summary of the IPv4 CEF table for the node on which the command is issued.

**Task ID**

| Task ID | Operations |
|---|---|
| cef | read |

**Examples**

The following sample output is from the **show cef ipv4 summary** command:

```
RP/0/RP0/CPU0:router# show cef ipv4 summary
Router ID is
10
0
.0.0.0

IP CEF with switching (Table Version 0)

  Load balancing: L3
  Tableid 0xe0000000,  Flags 0x301
  Refcount 367
  193 routes, 0 reresolve, 0 unresolved (0 old, 0 new), 13896 bytes
  204 load sharing elements, 51904 bytes, 154 references
  17 shared load sharing elements, 5536 bytes
  187 exclusive load sharing elements, 46368 bytes
  0 CEF route update drops, 175 revisions of existing leaves
  Resolution Timer: 15s
  0 prefixes modified in place
  0 deleted stale prefixes
  16 prefixes with label imposition, 51 prefixes with label information
Adjacency Table has 44 adjacencies
  1 incomplete adjacency
```

This table describes the significant fields shown in the display.

***Table 21: show cef ipv4 summary Command Field Descriptions***

| Field | Description |
|---|---|
| Load balancing | Current load-balancing mode. The default value is L3. |
| Table Version | Version of the CEF table. |
| tableid | Table identification number. |
| flags | Option value for the table |
| routes | Total number of routes. |
| reresolve | Total number of routes being reresolved. |
| unresolved (*x* old, *x* new) | Number of routes not yet resolved. |
| load sharing elements | Total number of internal load-sharing data structures. |
| bytes | Total memory used by internal load sharing data structures. |
| references | Total reference count of all internal load sharing data structures. |
| CEF resets | Number of CEF table resets. |
| revisions of existing leaves | Number of updates to existing prefixes. |
| Exponential (currently *x*s, peak *x*s) | Currently not used. |
| prefixes modified in place | Prefixes modified in place. |
| Adjacency Table has *x* adjacencies | Total number of adjacencies. |
| *x* incomplete adjacency | Total number of incomplete adjacencies. |

## Related Commands

| Command | Description |
|---|---|
| bundle-hash | Displays the path a bundle flow that comprises a source and destination address would take. |

# show cef ipv4 unresolved

To display unresolved routes in the IPv4 Cisco Express Forwarding (CEF) table, use the **show cef ipv4 unresolved** command in XR EXEC mode.

**show cef ipv4 unresolved** [**detail**] [**hardware** {**egress** | **ingress**}] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| detail | (Optional) Displays detailed information unresolved routes listed in the IPv4 CEF table. |
| hardware | (Optional) Displays detailed information about hardware. |
| egress | (Optional) Displays egress packet switch exchange (PSE). |
| ingress | (Optional) Displays ingress packet switch exchange (PSE). |
| **location** *node-id* | (Optional) Displays the unresolved routes in the IPv4 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, the output displays the unresolved routes for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show cef ipv4 unresolved** command when an unresolved route is detected:

```
RP/0/RP0/CPU0:router# show cef ipv4 unresolved

Prefix              Next Hop            Interface
10.3.3.3              102.2.2.2               ?
```

This table describes the significant fields shown in the display.

***Table 22: show cef ipv4 unresolved Command Field Descriptions***

| Field | Description |
|---|---|
| Prefix | Prefix of the unresolved CEF. |

| Field | Description |
|---|---|
| Next Hop | Next hop of the unresolved CEF. |
| Interface | Next hop interface. A question mark (?) indicates that the interface has not been resolved. |

# show cef ipv6

To display the IPv6 Cisco Express Forwarding (CEF) table, use the **show cef ipv6** command in XR EXEC mode.

**show cef** [ ] **ipv6** [*interface-type interface-number* | *ipv6-prefix/ prefix-length*] [**detail**] [**location***node-id*]

## Syntax Description

| | |
|---|---|
| interface-type interface-number | (Optional) IPv6 prefixes going through the specified next hop interface. |
| ipv6-prefix/prefix-length | (Optional) Longest prefix entry in the CEF table matching the specified IPv6 prefix and prefix length. |
| detail | (Optional) Displays detailed IPv6 CEF table information. |
| **location** *node-id* | (Optional) Displays the IPv6 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays the IPv6 CEF table for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

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

```
RP/0/RP0/CPU0:router# show cef ipv6

::/0

::/128
  drop
::1/128
  loopback
66::4/128
  receive    Loopback0
2222::/64
  connected  HundredGigE0/4/0/0
2222::1/128
  receive    HundredGigE0/4/0/0
3333::/64
  connected  HundredGigE0/3/0/0
3333::2/128
```

```
  receive    HundredGigE0/3/0/0
5656::2/128
  recursive  fe80::3031:48ff:fe53:5533, HundredGigE0/3/0/0
7777::/64
  connected  HundredGigE0/0/0/0
7777::2/128
  receive    HundredGigE0/0/0/0
9999::1/128
  recursive  fe80::205:5fff:fe1d:7600, HundredGigE0/4/0/0
ff00::/8
  drop
ff02::1/128
  receive
ff02::2/128
  receive
ff02::5/128
  receive
ff02::6/128
  receive
ff02::1:ff00:0/104
  receive
```

This table describes the significant fields shown in the display.

***Table 23: show cef ipv6 Command Field Descriptions***

| Field | Description |
|---|---|
| drop | Indicates that packets sent to the destination prefix are dropped. |
| loopback | Indicates that the prefix points to a loopback address. Packets sent to loopback addresses are dropped. |
| receive | Indicates that the prefix is configured on one of the router interfaces. Packets sent to those prefixes are received by the router. |
| connected | Indicates that the prefix points to a directly connected next-hop interface. |
| recursive | Indicates that the prefix is not directly connected but is reachable through the next-hop prefix displayed. |

The following sample output is from the **show cef ipv6** with the **detail** keyword:

```
RP/0/RP0/CPU0:router# show cef ipv6 detail


::/0
  flags: source_rib
  Loadinfo owner: <this route>
  fast adj: glean
  path 1:
    flags     :
    next hop  : ::
    interface :
HundredGigE/0/0/0

::/128
  flags: drop, source_fib
  Loadinfo owner: <this route>
  fast adj: drop
```

```
  path 1:
    flags     :
    next hop  : ::
    interface : <not specified>

::1/128
  flags: loopback, source_fib
  Loadinfo owner: <this route>
  fast adj: loopback
  path 1:
    flags     :
    next hop  : ::
    interface : <not specified>

66::4/128
  flags: receive, source_rib
  Loadinfo owner: <this route>
  fast adj: receive
  path 1:
    flags     : point-to-point
    next hop  : ::
    interface : Loopback0
```

This table describes the significant output fields shown in the display.

***Table 24: show cef ipv6 detail Command Field Descriptions***

| Field | Description |
|---|---|
| flags: | Properties of the indicated prefix. |
| Loadinfo owner: | Owner of the Loadinfo used by the prefix for forwarding. The Loadinfo owner is the prefix that owns the array of pointers to adjacencies. |
| fast adj: | Cached adjacency used for forwarding. |
| path 1: | The following three items are displayed below path 1:<br>• flags–Properties of the path.<br>• next hop–Next-hop prefix if the packet is being forwarded.<br>• interface–Next-hop interface if the packet is being forwarded. |

# show cef ipv6 adjacency

To display Cisco Express Forwarding (CEF) IPv6 adjacency status and configuration information, use the **show cef ipv6 adjacency** command in XR EXEC mode.

**show cef ipv6 adjacency** [*interface-type interface-path-id*] [**location** *node-id*] [**detail**] [**discard**] [**glean**] [**null**] [**punt**] [**remote**]

## Syntax Description

| | |
|---|---|
| interface-type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface- path-id* | (Optional) Either a physical interface instance or a virtual interface instance:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash mark between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| detail | (Optional) Displays the detailed adjacency information. |
| discard | (Optional) Filters out and displays only the discarded adjacency information. |
| glean | (Optional) Filters out and displays only the glean adjacency information. |
| null | (Optional) Filters out and displays only the null adjacency information. |
| punt | (Optional) Filters out and displays only the punt adjacency information. |
| remote | (Optional) Filters out and displays only the remote adjacency information. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays the CEF adjacency table for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

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

```
RP/0/RP0/CPU0:router# show cef ipv6 adjacency
```

This is a sample output from the **show cef ipv6 adjacency remote detail** command:

```
RP/0/RP0/CPU0:router# show cef ipv6 adjacency remote detail location 0/3/CPU0

Display protocol is ipv6
Interface    Address                                          Type    Refcount

Te0/2/0/3    Ifhandle: 0x8000240                              remote  2
             Adjacency: PT:0xa1bed9e4
             Interface: Te0/2/0/3
             Interface Type: 0x0, Base Flags: 0x0 (0xa55f3114)
             Nhinfo PT: 0xa55f3114, Idb PT: 0xa2d850d8, If Handle: 0x8000240
             Ancestor If Handle: 0x0


tt103        Ifhandle: 0x120                                  remote  1
             no next-hop adj
             Interface: NULLIFHNDL
             tunnel adjacency
             Interface Type: 0x24, Base Flags: 0x200 (0xa61ddc30)
             Nhinfo PT: 0xa61ddc30, Idb PT: 0xa2d851d8, If Handle: 0x120
             Ancestor If Handle: 0x0


tt2993       Ifhandle: 0xf9a0                                 remote  1
             no next-hop adj
             Interface: NULLIFHNDL
             tunnel adjacency
             Interface Type: 0x24, Base Flags: 0x200 (0xa65634f0)
             Nhinfo PT: 0xa65634f0, Idb PT: 0xa2d94a58, If Handle: 0xf9a0
             Ancestor If Handle: 0x0


tt2994       Ifhandle: 0xf9e0                                 remote  1
             no next-hop adj
             Interface: NULLIFHNDL
             tunnel adjacency
             Interface Type: 0x24, Base Flags: 0x200 (0xa65641e0)
             Nhinfo PT: 0xa65641e0, Idb PT: 0xa2d94a98, If Handle: 0xf9e0
```

```
                Ancestor If Handle: 0x0


tt2995          Ifhandle: 0xfa20                                remote  1
                no next-hop adj
                Interface: NULLIFHNDL
                tunnel adjacency
                Interface Type: 0x24, Base Flags: 0x200 (0xa6564350)
                Nhinfo PT: 0xa6564350, Idb PT: 0xa2d94ad8, If Handle: 0xfa20
                Ancestor If Handle: 0x0
```

# show cef ipv6 adjacency hardware

To display Cisco Express Forwarding (CEF) IPv6 adjacency hardware status and configuration information, use the **show cef ipv6 adjacency hardware** command in XR EXEC mode.

**show cef ipv6 adjacency hardware** {**egress** | **ingress**} [{**detail** | **discard** | **drop** | **glean** | **location** *node-id* | **null** | **punt** | **remote**}]

## Syntax Description

| | |
|---|---|
| egress | Displays information from the egress packet switch exchange (PSE) file. |
| ingress | Displays information from the ingress packet switch exchange (PSE) file. |
| detail | (Optional) Displays full details. |
| discard | (Optional) Displays the discard adjacency information. |
| drop | (Optional) Displays the drop adjacency information. |
| glean | (Optional) Displays the glean adjacency information. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| null | (Optional) Displays the null adjacency information. |
| punt | (Optional) Displays the punt adjacency information. |
| remote | (Optional) Displays the remote adjacency information. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output is from the **show cef ipv6 adjacency hardware** command:

```
RP/0/RP0/CPU0:router# show cef ipv6 adjacency hardware
```

# show cef ipv6 drops

To display IPv6 Cisco Express Forwarding (CEF) table packet drop counters, use the **show cef ipv6 drops** command in XR EXEC mode.

**show cef ipv6 drops** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays IPv6 CEF table packet drop counters for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

A packet might be dropped by the IPv6 CEF table because of unresolved CEF entries, unsupported features, absence of route information, absence of adjacency information, or an IP checksum error.

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays the packet drops for all nodes.

**Note** Because no hardware forwarding occurs on the route processor (RP), no packet drop information is displayed for that node.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

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

```
RP/0/RP0/CPU0:router# show cef ipv6 drops location 0/2/CPU0

 IPv6 CEF Drop Statistics
  Line status down      ingress :              0 egress :  Not Applicable
  Packet sanity fail    ingress :              0 egress :               0
  PLU set to drop       ingress :              0 egress :               0
  Unknown type,plu drop ingress :              0 egress :               0
  Packet length err     ingress :              0 egress :               0
  TCAM src-comp err     ingress :              0 egress :               0
```

This table describes the significant fields shown in the display.

*Table 25: show cef ipv6 drop Command Field Descriptions*

| Field | Description |
|---|---|
| Line status down | Packet drops due to the line protocol of the incoming interface being down. |
| Packet sanity fail | Packet drops due to the prefix failing the IPv6 sanity test. The sanity test verifies that the IPv6 packet is valid. |
| PLU set to drop | Packet drops due the IPv6 destination prefix being set to drop. |
| Unknown type, plu drop | Packet drops due to the prefix being of an unknown type. |
| Packet length errs | Length specified in the header does not match the actual length of the packet received. |
| TCAM src-comp err | Packet drops due to source compression errors that have occurred in the hardware. |

## Related Commands

| Command | Description |
|---|---|
| clear cef ipv6 drops, on page 111 | Clears IPv6 CEF packet drop counters. |

# show cef ipv6 exact-route

To display the path an IPv6 flow comprising a source and destination address would take, use the **show cef ipv6 exact-route** command in XR EXEC mode.

**show cef ipv6 exact-route** {*source-address destination-address* } [**protocol** *protocol-name* **source-port** *source-port* **destination-port** *destination-port* **ingress-interface** *typeinterface-path-id*] [**policy-class** *policy-class-value*] [**detail** | **location** *node-id*]

## Syntax Description

| | |
|---|---|
| *source-address* | The IPv6 source address in x:x::x format. |
| destination-address | The IPv6 destination address in x:x::x format. |
| **protocol** *protocol name* | (Optional) Displays the specified protocol for the route. |
| **source-port** *source-port* | (Optional) Sets the UDP source port. The range is from 0 to 65535. |
| **destination-port** *destination-port* | (Optional) Sets the UDP destination port. The range is from 0 to 65535. |
| ingress-interface | (Optional) Sets the ingress interface. |
| type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **policy-class** *value* | (Optional) Displays the class for the policy-based tunnel selection. The range for the tunnel policy class value is from 1 to 7. |
| detail | (Optional) Displays full CEF entry information. |
| **location** *node-id* | (Optional) Displays the IPv6 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If the Layer 4 information is enabled, the source-port, destination-port, protocol, and ingress-interface fields are required. Otherwise, the output of the **show cef ipv6 exact-route** command is not correct.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output is from the **show cef ipv6 exact-route** command:

```
RP/0/RP0/CPU0:router# show cef ipv6 exact-route 222::2 9999::6751 location

0/3/CPU0 source address: 222::2 destination address: 9999::6751
interface : HundredGigE0/3/0/3 non local interface
```

# show cef ipv6 exceptions

To display IPv6 Cisco Express Forwarding (CEF) exception packet counters, use the **show cef ipv6 exceptions** command in XR EXEC mode.

**show cef ipv6 exceptions** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays IPv6 CEF exception packet counters for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

CEF exception packets are those packets that have been sent from the hardware to the software because they require additional handling. The types of IPv6 CEF exception packets are displayed in the output of **show cef ipv6 exceptions**.

If you do not specify a node with **location** keyword and *node-id* argument, this command displays IPv6 CEF exception packet counters for all nodes.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

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

```
RP/0/RP0/CPU0:router# show cef ipv6 exceptions location 0/3/CPU0

IPv6 CEF Exception Statistics
Node: 0/3/CPU0
  TTL err               ingress :               0 egress :  Not Applicable
  Link-local dst addr   ingress :               0 egress :               0
  Hop-by-Hop header     ingress :               0 egress :               0
  PLU entry set to punt ingress :               0 egress :               0
  Packet too big        ingress :  Not Applicable egress :               0
  Med priority punt     ingress :               0 egress :  Not Applicable
```

This table describes the significant fields shown in the display.

***Table 26: show cef ipv6 exceptions Command Field Descriptions***

| Field | Description |
|---|---|
| TTL err | Packets sent to software for processing because the packet header of the IPv6 prefix had a TTL[6] error. |

| Field | Description |
|---|---|
| Link-local dst addr | Packets sent to the software for processing because the destination address of the IPv6 prefix is link local. |
| Hop-by-Hop header | Packets sent to the software for processing because the IPv6 packet has a hop-by-hop header. |
| PLU entry set to punt | Packets sent to software for processing because the IPv6 prefix is set to punt. |
| Packet too big | Packets sent to the software for processing because the packet size exceeded the MTU[7]. |
| Med priority punt | Field used internally for troubleshooting. |

[6] TTL = time to live
[7] MTU = maximum transmission unit

## Related Commands

| Command | Description |
|---|---|
| clear cef ipv6 exceptions, on page 113 | Clears IPv6 CEF exception packet counters. |

# show cef ipv6 hardware

To display Cisco Express Forwarding (CEF) IPv6 hardware status and configuration information, use the **show cef ipv6 hardware** command in XR EXEC mode.

**show cef ipv6 hardware** {**egress** | **ingress**}[{**detail** | **location** *node-id*}]

## Syntax Description

| | |
|---|---|
| egress | Displays information from the egress packet switch exchange (PSE) file. |
| ingress | Displays information from the ingress packet switch exchange (PSE) file. |
| detail | (Optional) Displays full details. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output displays the full details from the **show cef ipv6 hardware** command:

```
RP/0/RP0/CPU0:router# show cef ipv6 hardware egress detail

::/0, version 0, proxy default, default route handler, drop adjacency, internal
 Prefix Len 0, traffic index 0, precedence routine (0)
  gateway array (0x0) reference count 1, flags 0x4000, source 4,
                [0 type 3 flags 0x109000 (0x7895114c) ext 0x0 (0x0)]
  LW-LDI[type=3, refc=1, ptr=0x78a7d0dc, sh-ldi=0x7895114c]
   via point2point, 0 dependencies, weight 0, class 0
    next hop point2point
     drop adjacency


    Load distribution: 0 (refcount 0)

    Hash  OK  Interface                 Address
    0     Y   Unknown                   drop
ff02::/16, version 0, receive
  Prefix Len 16
ff02::2/128, version 0, receive
```

```
  Prefix Len 128
ff02::1:ff00:0/104, version 0, receive
  Prefix Len 104
```

# show cef ipv6 interface

To display IPv6 Cisco Express Forwarding (CEF)-related information for an interface, use the **show cef ipv6 interface** command in XR EXEC mode.

**show cef ipv6 interface** *type interface-path-id* [**detail**] [**location** *node-id*] **[rpf-drop]**

## Syntax Description

| | |
|---|---|
| type | Interface type. For more information, use the question mark (?) online help function. |
| interface-path-id | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| detail | (Optional) Displays detailed CEF information for all the interfaces on the node in which the command is issued. |
| **location** *node-id* | (Optional) Displays IPv4 CEF-related information for an interface. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **rpf-drop** | (Optional) Displays information about the drops due to IPv6 unicast RPF. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, the **show cef ipv6 interface** command displays the CEF-related information for the interface on the route processor.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

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

```
RP/0/RP0/CPU0:router# show cef ipv6 interface
```

# show cef ipv6 interface bgp-policy-statistics

To display IPv6 Cisco Express Forwarding (CEF)-related BGP policy statistics information for an interface, use the **show cef ipv6 interface bgp-policy-statistics** command in XR EXEC mode.

**show cef ipv6 interface** *type interface-path-id* **bgp-policy-statistics** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| type | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **show cef ipv6 interface bgp-policy-statistics** command displays all the configured BGP policy counters for the specified interface.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output is from the **show cef ipv6 interface bgp-policy-statistics** command:

```
RP/0/RP0/CPU0:router# show cef ipv6 interface bgp-policy-statistics
```

# show cef ipv6 non-recursive

To display the IPv6 nonrecursive prefix entries in the IPv6 Cisco Express Forwarding (CEF) table, use the **show cef ipv6 non-recursive** command in XR EXEC mode.

**show cef ipv6 non-recursive** [**hardware** {**egress** | **ingress**}] [**detail**] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| hardware | (Optional) Displays Cisco Express Forwarding (CEF) IPv6 hardware status and configuration information. |
| egress | (Optional) Displays information from the egress packet switch exchange (PSE) file. |
| ingress | (Optional) Displays information from the ingress packet switch exchange (PSE) file. |
| detail | (Optional) Displays full details. |
| **location** *node-id* | (Optional) Displays the nonrecursive prefix entries in the IPv6 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays the nonrecursive routes for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following is sample output from the **show cef ipv6 non-recursive** command:

```
RP/0/RP0/CPU0:router# show cef ipv6 non-recursive

::/0

::/128
  drop
::1/128
  loopback
66::4/128
  receive    Loopback0
2222::/64
  connected  HundredGigE0/4/0/0
2222::1/128
  receive    HundredGigE0/4/0/0
3333::/64
```

```
  connected  HundredGigE0/3/0/0
3333::2/128
  receive    HundredGigE0/3/0/0
7777::/64
  connected  HundredGigE0/0/0/0
7777::2/128
  receive    HundredGigE0/0/0/0
ff00::/8
  drop
ff02::1/128
  receive
ff02::2/128
  receive
ff02::5/128
  receive
ff02::6/128
  receive
ff02::1:ff00:0/104
  receive
```

This table describes the significant fields shown in the display.

***Table 27: show cef ipv6 non-recursive Command Field Descriptions***

| Field | Description |
|---|---|
| drop | Indicates that packets sent to the destination prefix are dropped. |
| loopback | Indicates that the prefix points to a loopback address. Packets sent to loopback addresses are dropped. |
| receive | Indicates that the prefix is configured on one of the router interfaces. Packets sent to those prefixes are received by the router. |
| connected | Indicates that the prefix points to a directly connected next-hop interface. |

# show cef ipv6 resource

To display the IPv6 nonrecursive prefix entries in the IPv6 Cisco Express Forwarding (CEF) table, use the **show cef ipv6 resource** command in XR EXEC mode.

**show cef ipv6 resource** [**detail**] [**hardware** {**egress** | **ingress**}] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| detail | (Optional) Displays detailed information resources listed in the IPv6 CEF table. |
| hardware | (Optional) Displays Cisco Express Forwarding (CEF) IPv6 hardware status and configuration information. |
| egress | (Optional) Displays information from the egress packet switch exchange (PSE) file. |
| ingress | (Optional) Displays information from the ingress packet switch exchange (PSE) file. |
| **location** *node-id* | (Optional) Displays the IPv6 resource entries in the IPv6 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, the output displays the IPv6 CEF nonrecursive routes for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

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

```
RP/0/RP0/CPU0:router# show cef ipv6 resource

CEF resource availability summary state: GREEN
  ipv4 shared memory resource: GREEN
  ipv6 shared memory resource: GREEN
  mpls shared memory resource: GREEN
  common shared memory resource: GREEN
  TABLE hardware resource: GREEN
  LEAF hardware resource: GREEN
  LOADINFO hardware resource: GREEN
  NHINFO hardware resource: GREEN
  LABEL_INFO hardware resource: GREEN
  IDB hardware resource: GREEN
  FRR_NHINFO hardware resource: GREEN
```

```
LDSH_ARRAY hardware resource: GREEN
RSRC_MON hardware resource: GREEN
```

# show cef ipv6 summary

To display a summary of the IPv6 Cisco Express Forwarding (CEF) table, use the **show cef ipv6 summary** command in XR EXEC mode.

**show cef ipv6 summary** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays a summary of the IPv6 CEF table for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays a summary of the IPv6 CEF table for the node on which the command is issued.

**Task ID**

| Task ID | Operations |
|---|---|
| cef | read |

**Examples**

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

```
RP/0/RP0/CPU0:router# show cef ipv6 summary

IP CEF with switching (Table Version 0)

  Load balancing: L3
  Tableid 0xe0800000, Flags 0x301
  Refcount 12
  4 routes, 0 reresolve, 0 unresolved (0 old, 0 new), 288 bytes
  0 load sharing elements, 0 bytes, 0 references
  0 shared load sharing elements, 0 bytes
  0 exclusive load sharing elements, 0 bytes
  0 CEF route update drops, 0 revisions of existing leaves
  Resolution Timer: 15s
  0 prefixes modified in place
  0 deleted stale prefixes
  0 prefixes with label imposition, 0 prefixes with label information
Adjacency Table has 44 adjacencies
  1 incomplete adjacency
```

This table describes the significant fields shown in the display.

***Table 28: show cef ipv6 summary Command Field Descriptions***

| Field | Description |
|---|---|
| Load balancing | Current load-balancing mode. The default value is L3. |
| Table Version | Version of the CEF table. |
| routes | Total number of routes. |
| unresolved (*x* old, *x* new) | Number of routes not yet resolved. |
| load sharing elements | Total number of internal load-sharing data structures. |
| bytes | Total memory used by internal load sharing data structures. |
| references | Total reference count of all internal load sharing data structures. |
| CEF resets | Number of CEF table resets. |
| revisions of existing leaves | Number of updates to existing prefixes. |
| Exponential (currently *x*s, peak *x*s) | Currently not used. |
| prefixes modified in place | Prefixes modified in place. |
| Router ID | Router identification. |
| Adjacency Table has *x* adjacencies | Total number of adjacencies. |
| *x* incomplete adjacency | Total number of incomplete adjacencies. |

## Related Commands

| Command | Description |
|---|---|
| bundle-hash | Displays the path a bundle flow that comprises a source and destination address would take. For more information, see *Interface and Hardware Component Command Reference for the Cisco NCS 6000 Series Routers* |
| cef load-balancing fields, on page 100 | Selects the hashing algorithm that is used for load balancing when forwarding. |

# show cef ipv6 unresolved

To display the unresolved routes in the IPv6 Cisco Express Forwarding (CEF) table, use the **show cef ipv6 unresolved** command in XR EXEC mode.

**show cef ipv6 unresolved** [**detail**] [**hardware** {**egress** | **ingress**}] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **detail hardware** | (Optional) Displays full details.<br>(Optional) Displays Cisco Express Forwarding (CEF) IPv6 hardware status and configuration information. |
| **hardware egress** | (Optional) Displays Cisco Express Forwarding information from the egress packet switch exchange ( CEF PSE ) IPv6 hardware status and configuration information file . |
| **egress ingress** | (Optional) Displays information from the egress ingress packet switch exchange (PSE) file. |
| **ingress detail** | (Optional) Displays information from the ingress packet switch exchange (PSE) file full details . |
| **location** *node-id* | (Optional) Displays the unresolved routes in the IPv6 CEF table for the specified node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, this command displays the unresolved routes for the node on which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

This following is sample output from **show cef ipv6 unresolved** command when an unresolved route is detected:

```
RP/0/RP0/CPU0:router# show cef ipv6 unresolved

9999::/64
  unresolved
```

This table describes the significant fields shown in the display.

***Table 29: show cef ipv6 unresolved Command Field Descriptions***

| Field | Description |
|---|---|
| *xxxx*::/*xx* | Detected unresolved route. |

# show cef mpls adjacency

To display the Multiprotocol Label Switching (MPLS) adjacency table, use the **show cef mpls adjacency** command in XR EXEC mode.

**show cef mpls adjacency** [*interface-type interface-path-id*] [{**detail** | **discard** | **drop** | **glean** | **null** | **punt** | **remote**}] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| interface-type | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface- path-id* | (Optional) Either a physical interface instance or a virtual interface instance:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash mark between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| detail | (Optional) Displays full details. |
| discard | (Optional) Displays the discard adjacency information. |
| drop | (Optional) Displays the drop adjacency information. |
| glean | (Optional) Displays the glean adjacency information. |
| null | (Optional) Displays the null adjacency information. |
| punt | (Optional) Displays the punt adjacency information. |
| remote | (Optional) Displays the remote adjacency information. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, the **show cef mpls adjacency** command displays the MPLS adjacency table for the node in which the command is issued.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

This following is sample output from **show cef mpls adjacency** command:

```
RP/0/RP0/CPU0:router# show cef mpls adjacency
```

## Related Commands

| Command | Description |
|---|---|
| show cef mpls adjacency hardware, on page 187 | Displays the Multiprotocol Label Switching (MPLS) adjacency hardware status and configuration information. |
| show cef mpls interface, on page 189 | Displays the Multiprotocol Label Switching (MPLS) Cisco Express Forwarding (CEF)-related information for an interface. |
| show cef mpls unresolved, on page 191 | Displays the Multiprotocol Label Switching (MPLS) unresolved routes. |

# show cef mpls adjacency hardware

To display the Multiprotocol Label Switching (MPLS) adjacency hardware status and configuration information, use the **show cef mpls adjacency hardware** command in XR EXEC mode.

**show cef mpls adjacency hardware** {**egress** | **ingress**} [{**detail** | **discard** | **drop** | **glean** | **location** *node-id* | **null** | **punt** | **remote**}]

## Syntax Description

| | |
|---|---|
| egress | Displays information from the egress packet switch exchange (PSE) file. |
| ingress | Displays information from the ingress packet switch exchange (PSE) file. |
| detail | (Optional) Displays full details. |
| discard | (Optional) Displays the discard adjacency information. |
| drop | (Optional) Displays the drop adjacency information. |
| glean | (Optional) Displays the glean adjacency information. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| null | (Optional) Displays the null adjacency information. |
| punt | (Optional) Displays the punt adjacency information. |
| remote | (Optional) Displays the remote adjacency information. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

This following is sample output from **show cef mpls adjacency hardware** command:

```
RP/0/RP0/CPU0:router# show cef mpls adjacency hardware
```

## Related Commands

| Command | Description |
|---|---|
| show cef mpls adjacency, on page 185 | Displays the Multiprotocol Label Switching (MPLS) adjacency table. |
| show cef mpls interface, on page 189 | Displays the Multiprotocol Label Switching (MPLS) Cisco Express Forwarding (CEF)-related information for an interface. |
| show cef mpls unresolved, on page 191 | Displays the Multiprotocol Label Switching (MPLS) unresolved routes. |

# show cef mpls interface

To display the Multiprotocol Label Switching (MPLS) Cisco Express Forwarding (CEF)-related information for an interface, use the **show cef mpls interface** command in XR EXEC mode.

**show cef mpls interface** [*type interface-path-id*] [**detail**] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| type | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the modular services card or line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( RP0 or RP1 ) and the module is CPU0. Example: interface MgmtEth0/ RP1 /CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| detail | (Optional) Displays detailed CEF information for all the interfaces on the node in which the command is issued. |
| **location** *node-id* | (Optional) Displays IPv4 CEF-related information for an interface. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If you do not specify a node with the **location** keyword and *node-id* argument, the **show cef mpls interface** command displays the CEF-related information for the interface on the route processor.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output is from the **show cef mpls interface** command:

```
RP/0/RP0/CPU0:router# show cef mpls interface
```

## Related Commands

| Command | Description |
|---|---|
| show cef mpls adjacency, on page 185 | Displays the Multiprotocol Label Switching (MPLS) adjacency table. |
| show cef mpls adjacency hardware, on page 187 | Displays the Multiprotocol Label Switching (MPLS) adjacency hardware status and configuration information. |
| show cef mpls unresolved, on page 191 | Displays the Multiprotocol Label Switching (MPLS) unresolved routes. |

# show cef mpls unresolved

To display the Multiprotocol Label Switching (MPLS) unresolved routes, use the **show cef mpls unresolved** command in XR EXEC mode.

**show cef mpls unresolved** [**detail**] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| detail | (Optional) Displays detailed adjacency information, including Layer 2 information. |
| **location** *node-id* | (Optional) Displays detailed CEF information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| cef | read |

## Examples

The following sample output is from the **show cef mpls unresolved** command:

```
RP/0/RP0/CPU0:router# show cef mpls unresolved

Label/EOS           Next Hop            Interface
20001/0
20001/1
```

This table describes the significant fields shown in the display.

***Table 30: show cef mpls unresolved Command Field Descriptions***

| Field | Description |
|---|---|
| Label/EOS | MPLS forwarding label/End of Stack (EOS) bit. |
| Next Hop | Next hop of the prefix. |
| Interface | Interface associated with the prefix. |

## Related Commands

| Command | Description |
|---|---|
| show cef mpls adjacency, on page 185 | Displays the Multiprotocol Label Switching (MPLS) adjacency table. |
| show cef mpls adjacency hardware, on page 187 | Displays the Multiprotocol Label Switching (MPLS) adjacency hardware status and configuration information. |
| show cef mpls interface, on page 189 | Displays the Multiprotocol Label Switching (MPLS) Cisco Express Forwarding (CEF)-related information for an interface. |

# DHCP Commands

This chapter describes the Cisco IOS XR software commands used to configure and monitor Dynamic Host Configuration Protocol (DHCP).

For detailed information about DHCP concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# broadcast-flag policy check

To configure Dynamic Host Configuration Protocol (DHCP) IPv4 Relay to broadcast only BOOTREPLY packets if the DHCP IPv4 broadacst flag is set in the DHCP IPv4 header, use the **broadcast-flag policy check** command in DHCP IPv4 relay profile configuration submode . By default, the DHCP IPv4 Relay always broadcasts BOOTREPLY packets. To restore the default, use the **no** form of this command.

**broadcast-flag policy** { **check** }

## Syntax Description

| | |
|---|---|
| **check** | Checks the broadcast flag in packets. |
| **unicast-always** | Sets the broadcast-flag policy to unicast-always. |

## Command Default

Relay agent always broadcasts DHCP IPv4 packets to a client.

## Command Modes

DHCP IPv4 relay profile configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

This an example of the **broadcast-flag policy check** command:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)# profile client relay
RP/0/RP0/CPU0:router(config-dhcpv4-relay-profile)# broadcast-flag policy check
```

## Related Commands

| Command | Description |
|---|---|
| dhcp ipv4 , on page 196 | Enables DHCP for IPv4 and enters DHCP IPv4 configuration mode. |
| helper-address | Configures the DHCP relay agent to relay packets to a specific DHCP server. |

| Command | Description |
|---|---|
| relay information check , on page 203 | Configures a DHCP server to validate the relay agent information option in forwarded BOOTREPLY messages. |
| relay information option , on page 205 | Enables the system to insert a DHCP relay agent information option in forwarded BOOTREQUEST messages to a DHCP server. |
| relay information option allow-untrusted , on page 207 | Configures the DHCP component to not drop BOOTREQUEST messages that have the relay information option set and the giaddr set to zero. |
| relay information policy , on page 209 | Configures how a relay agent processes BOOTREQUEST messages that already contain a relay information option. |

# dhcp ipv4

To enable Dynamic Host Configuration Protocol (DHCP) for IPv4 and to enter DHCP IPv4 configuration mode, use the **dhcp ipv4** command in XR Config mode. To disable DHCP for IPv4 and exit the DHCP IPv4 configuration mode, use the **no** form of this command.

**dhcp ipv4**

## Syntax Description

This command has no keywords or arguments.

## Command Modes

None

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **dhcp ipv4** command to enter DHCP IPv4 configuration mode.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

This example shows how to enable DHCP for IPv4:

```
RP/0/RP0/CPU0:router# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)#
```

# giaddr policy

To configure how Dynamic Host Configuration Protocol (DHCP) IPv4 Relay processes BOOTREQUEST packets that already contain a nonzero giaddr attribute, use the **giaddr policy** command in DHCP IPv4 profile relay configuration submode. To restore the default giaddr policy, use the **no** form of this command.

**giaddr policy** {**replace** | **drop**}
**no giaddr policy** {**replace** | **drop**}

## Syntax Description

| | |
|---|---|
| replace | Replaces the existing giaddr value with a value that it generates. |
| drop | Drops the packet that has an existing nonzero giaddr value. |

## Command Default

DHCP IPv4 relay retains the existing nonzero giaddr value in the DHCP IPv4 packet received from a client value.

## Command Modes

DHCP IPv4 profile relay configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **giaddr policy** command affects only the packets that are received from a DHCP IPv4 client that have a nonzero giaddr attribute.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to use the **giaddr policy** command:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)# profile client relay
RP/0/RP0/CPU0:router(config-dhcpv4-relay-profile)# giaddr policy drop
```

## Related Commands

| Command | Description |
|---|---|
| dhcp ipv4 , on page 196 | Enables DHCP for IPv4 and enters DHCP IPv4 configuration mode. |
| helper-address | Configures the DHCP relay agent to relay packets to a specific DHCP Server. |

| Command | Description |
|---|---|
| relay information check , on page 203 | Configures a DHCP server to validate the relay agent information option in forwarded BOOTREPLY messages. |
| relay information option , on page 205 | Enables the system to insert a DHCP relay agent information option in forwarded BOOTREQUEST messages to a DHCP server. |
| relay information option allow-untrusted , on page 207 | Configures the DHCP component to not drop BOOTREQUEST messages that have the relay information option set and the giaddr set to zero. |
| relay information policy , on page 209 | Configures how a relay agent processes BOOTREQUEST messages that already contain a relay information option. |

# interface (relay profile)

To configure a relay profile on an interface, use the **interface (relay profile)** command in Dynamic Host Configuration Protocol (DHCP) IPv4 configuration mode. To disable this feature, use the **no** form of the command.

**interface interface-type interface-path-id** {**none** | **relay**}
**no interface interface-type interface-path-id** {**none** | **relay**}

## Syntax Description

| | |
|---|---|
| interface-type | Interface type. For more information, use the question mark (?) online help function. |
| interface-path-id | Either a physical interface instance or a virtual interface instance. |
| none | Disables DHCP at the specified interface. |
| relay | Specifies a relay profile for the interface. |

## Command Modes

DHCP IPv4 configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to configure a relay profile on an interface:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)# interface HundredGigE 0/7/0/0
RP/0/RP0/CPU0:router(config-dhcpv4)# interface HundredGigE 0/7/0/0 relay profile client
```

## Related Commands

| Command | Description |
|---|---|
| broadcast-flag policy check , on page 194 | Configures a relay agent to only broadcast DHCP IPv4 BOOTREPLY messages to a client, if the DHCP IPv4 broadcast flag is set in the DHCP IPv4 header. |
| dhcp ipv4 , on page 196 | Enables Dynamic Host Configuration Protocol (DHCP) for IPv4 and enters DHCP IPv4 configuration mode. |

| Command | Description |
|---|---|
| relay information policy | Configures how a relay agent processes BOOTREQUEST messages that already contain a nonzero giaddr attribute. |
| helper-address | Configures the DHCP relay agent to relay packets to a specific DHCP Server. |
| relay information check , on page 203 | Configures a DHCP server to validate the relay agent information option in forwarded BOOTREPLY messages. |
| relay information option , on page 205 | Enables the system to insert a DHCP relay agent information option in forwarded BOOTREQUEST messages to a DHCP server. |
| relay information option allow-untrusted , on page 207 | Configures the DHCP component to not drop BOOTREQUEST messages that have the relay information option set and the giaddr set to zero. |
| relay information policy , on page 209 | Configures how a relay agent processes BOOTREQUEST messages that already contain a relay information option. |

# profile relay

To configure a relay profile for the Dynamic Host Configuration Protocol (DHCP) IPv4 component and to enter the profile relay mode, use the **profile relay** command in DHCP IPv4 configuration mode. To disable this feature and exit the profile relay mode, use the **no** form of this command.

**profile** *profile name* **relay**
**no profile** *profile name* **relay**

## Syntax Description

| | |
|---|---|
| profile name | Name that uniquely identifies the relay profile. |

## Command Modes

DHCP IPv4 configuration

W3

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to use the **profile relay** command:

```
RP/0/RP0/CPU0:router(config)# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)# profile client relay
```

## Related Commands

| Command | Description |
|---|---|
| broadcast-flag policy check , on page 194 | Configures a relay agent to only broadcast DHCP IPv4 BOOTREPLY messages to a client, if the DHCP IPv4 broadcast flag is set in the DHCP IPv4 header. |
| dhcp ipv4 , on page 196 | Enables DHCP for IPv4 and enters DHCP IPv4 configuration mode. |
| relay information policy | Configures how a relay agent processes BOOTREQUEST messages that already contain a nonzero giaddr attribute. |
| helper-address | Configures the DHCP relay agent to relay packets to a specific DHCP Server. |
| interface (relay profile), on page 199 | Specifies a relay profile on an interface. |

| Command | Description |
|---|---|
| relay information check , on page 203 | Configures a DHCP server to validate the relay agent information option in forwarded BOOTREPLY messages. |
| relay information option , on page 205 | Enables the system to insert a DHCP relay agent information option in forwarded BOOTREQUEST messages to a DHCP server. |
| relay information option allow-untrusted , on page 207 | Configures the DHCP component to not drop BOOTREQUEST messages that have the relay information option set and the giaddr set to zero. |
| relay information policy , on page 209 | Configures how a relay agent processes BOOTREQUEST messages that already contain a relay information option. |

# relay information check

To configure a Dynamic Host Configuration Protocol (DHCP) IPv4 Relay to validate the relay agent information option in forwarded BOOTREPLY messages, use the **relay information check** command in DHCP IPv4 relay profile configuration submode. To disable this feature, use the **no** form of this command.

**relay information check**

## Syntax Description

This command has no keywords or arguments.

## Command Default

DHCP validates the relay agent information option.

## Command Modes

DHCP IPv4 relay profile configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |
| basic-services | read, write |

This example shows how to use the **relay information check** command:

```
RP/0/RP0/CPU0:router#config
RP/0/RP0/CPU0:router(config)# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)# profile client relay
RP/0/RP0/CPU0:router(config-dhcpv4-relay-profile)# relay information check
```

## Related Commands

| Command | Description |
|---|---|
| dhcp ipv4 , on page 196 | Enables DHCP for IPv4 and enters DHCP IPv4 configuration mode. |
| helper-address | Configures the DHCP relay agent to relay packets to a specific DHCP Server. |

| Command | Description |
|---|---|
| relay information option , on page 205 | Enables the system to insert a DHCP relay agent information option in forwarded BOOTREQUEST messages to a DHCP server. |
| relay information option allow-untrusted , on page 207 | Configures the DHCP component to not drop BOOTREQUEST messages that have the relay information option set and the giaddr set to zero. |

# relay information option

To configure Dynamic Host Configuration Protocol (DHCP) IPv4 relay or DHCP snooping Relay to insert relay agent information option in forwarded BOOTREQUEST messages to a DHCP server, use the **relay information option** command in DHCP IPv4 relay profile relay configuration or DHCP IPv4 profile snoop submode. To disable inserting relay information into forwarded BOOTREQUEST messages, use the **no** form of this command.

**relay information option**

## Syntax Description

This command has no keywords or arguments.

## Command Default

None

## Command Modes

DHCP IPv4
relay
profile
relay
configuration

DHCP IPv4 profile snoop configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **relay information option** command automatically adds the circuit identifier suboption and the remote ID suboption to the DHCP relay agent information option.

The **relay information option** command enables a DHCP server to identify the user (for example, cable access router) sending the request and initiate appropriate action based on this information. By default, DHCP does not insert relay information.

If the **information option** command is enabled, DHCP snooping mode does not set the giaddr field in the DHCP packet.

The upstream DHCP server or DHCP relay interface must be configured to accept this type of packet using the **relay information option allow-untrusted** configuration. This configuration prevents the server or relay from dropping the DHCP message.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |
| basic-services | read, write |

This example shows how to use the **relay information option** command:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)# profile client relay
RP/0/RP0/CPU0:router(config-dhcpv4-relay-profile)# relay information option
```

## Related Commands

| Command | Description |
|---|---|
| dhcp ipv4 , on page 196 | Enables DHCP for IPv4 and enters DHCP IPv4 configuration mode. |
| helper-address | Configures the DHCP relay agent to relay packets to a specific DHCP Server. |
| relay information check , on page 203 | Configures a DHCP server to validate the relay agent information option in forwarded BOOTREPLY messages. |
| relay information option allow-untrusted , on page 207 | Configures the DHCP component to not drop BOOTREQUEST messages that have the relay information option set and the giaddr set to zero. |

# relay information option allow-untrusted

To configure the Dynamic Host Configuration Protocol (DHCP) IPv4 relay or DHCP snooping Relay not to drop discard BOOTREQUEST packets that have the relay information option set and the giaddr set to zero, use the **relay information option allow-untrusted** command in DHCP IPv4 relay profile configuration submode or DHCP IPv4 profile snoop configuration submode. To restore the default behavior, which is to discard the BOOTREQUEST packets that have the relay information option and set the giaddr set to zero, use the **no** form of this command.

**relay information option allow-untrusted**

## Syntax Description

This command has no keywords or arguments.

## Command Default

The packet is dropped if the relay information is set and the giaddr is set to zero.

## Command Modes

DHCP IPv4
relay
profile
relay
configuration

DHCP IPv4 profile snoop configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

According to RFC 3046, relay agents (and servers) receiving a DHCP packet from an untrusted circuit with giaddr set to zero but with a relay agent information option already present in the packet shall discard the packet and increment an error count. This configuration prevents the server or relay from dropping the DHCP message.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |
| basic-services | read, write |

## Examples

This example shows how to use the **relay information option allow-untrusted** command:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)# profile client relay
RP/0/RP0/CPU0:router(config-dhcpv4-relay-profile)# relay information option allow-untrusted
```

## Related Commands

| Command | Description |
|---|---|
| dhcp ipv4 , on page 196 | Enables DHCP for IPv4 and enters DHCP IPv4 configuration mode. |
| helper-address | Configures the DHCP relay agent to relay packets to a specific DHCP Server. |
| relay information check , on page 203 | Configures a DHCP server to validate the relay agent information option in forwarded BOOTREPLY messages. |
| relay information option , on page 205 | Enables the system to insert a DHCP relay agent information option in forwarded BOOTREQUEST messages to a DHCP server. |

# relay information policy

To configure how the Dynamic Host Configuration Protocol (DHCP) IPv4 relay processes BOOTREQUEST packets that already contain a relay information option, use the **relay information policy** command in DHCP IPv4 relay profile configuration submode. To restore the default relay information policy, use the **no** form of this command.

**relay information policy** {**drop** | **keep**}

## Syntax Description

| | |
|---|---|
| drop | Directs the DHCP IPv4 Relay to discard BOOTREQUEST packets with the existing relay information option. |
| keep | Directs the DHCP IPv4 Relay not to discard a BOOTREQUEST packet that is received with an existing relay information option and to keep the existing relay information option value. |

## Command Default

The DHCP IPv4 Relay does not discard a BOOTREQUEST packet that has an existing relay information option. The option and the existing relay information option value is replaced.

## Command Modes

DHCP IPv4 relay profile configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |
| basic-services | read, write |

## Examples

This is sample output from executing the **relay information policy** command:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# dhcp ipv4
RP/0/RP0/CPU0:router(config-dhcpv4)# profile client relay
RP/0/RP0/CPU0:router(config-dhcpv4-relay-profile)# relay information policy keep
```

## Related Commands

| Command | Description |
|---|---|
| dhcp ipv4 , on page 196 | Enables DHCP for IPv4 and enters DHCP IPv4 configuration mode. |
| helper-address | Configures the DHCP relay agent to relay packets to a specific DHCP Server. |

| Command | Description |
| --- | --- |
| relay information check , on page 203 | Configures a DHCP server to validate the relay agent information option in forwarded BOOTREPLY messages. |
| relay information option , on page 205 | Enables the system to insert a DHCP relay agent information option in forwarded BOOTREQUEST messages to a DHCP server. |
| relay information option allow-untrusted , on page 207 | Configures the DHCP component to not drop BOOTREQUEST messages that have the relay information option set and the giaddr set to zero. |

# show dhcp ipv4 relay profile

To display Dynamic Host Configuration Protocol (DHCP) relay agent status, use the **show dhcp ipv4 relay profile** command in XR EXEC mode.

**show dhcp ipv4 relay profile**

## Syntax Description

This command has no keywords or arguments.

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

This command displays the relay profiles created for DHCP IPv4.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read |

## Examples

The following is sample output from the **show dhcp ipv4 relay profile** command:

```
RP/0/RP0/CPU0:router# show dhcp ipv4 relay profile

DHCP IPv4 Relay Profiles
--------------------------
r1
r2
```

## Related Commands

| Command | Description |
|---|---|
| show dhcp ipv4 relay profile name | Displays Dynamic Host Configuration Protocol (DHCP) relay agent status, specific to a relay profile. |

# Host Services and Applications Commands

This chapter describes the commands used to configure and monitor host services and applications, such as Domain Name System (DNS), Telnet, File Transfer Protocol (FTP), Trivial File Transfer Protocol (TFTP).

For detailed information about host services and applications concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# cinetd rate-limit

To configure the rate limit at which service requests are accepted by Cisco inetd (Cinetd), use the **cinetd rate-limit** command in XR Config mode. To restore the default, use the **no** form of this command.

**cinetd rate-limit** *value*
**no cinetd rate-limit** *value*

## Syntax Description

| | |
|---|---|
| value | Number of service requests that are accepted per second. Range is 1 to 100. Default is 1. |

## Command Default

One service request per second is accepted.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Any service request that exceeds the rate limit is rejected. The rate limit is applied to individual applications.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows the **cinetd rate-limit** being set to 10:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# cinetd rate-limit 10
```

# clear host

To delete temporary entries from the hostname-to-address cache, use the **clear host** command in XR EXEC mode.

**clear host** {*host-name* | *****}

## Syntax Description

| | |
|---|---|
| host-name | Name of host to be deleted. |
| * | Specifies that all entries in the local cache be deleted. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The dynamic host entries in the cache are cleared.

The temporary entries in the cache are cleared; the permanent entries that were entered with the domain ipv4 host, on page 216 or the domain ipv6 host, on page 217 command are not cleared.

By default, no static mapping is configured.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | execute |

## Examples

The following example shows how to clear all temporary entries from the hostname-and-address cache:

```
RP/0/RP0/CPU0:router# clear host *
```

## Related Commands

| Command | Description |
|---|---|
| domain ipv4 host, on page 216 | Defines a static IPv4 hostname-to-address mapping in the host cache. |
| domain ipv6 host, on page 217 | Defines a static IPv6 hostname-to-address mapping in the host cache. |
| show hosts, on page 237 | Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses. |

# domain ipv4 host

To define a static hostname-to-address mapping in the host cache using IPv4, use the **domain ipv4 host** command in XR Config mode. To remove the **domain ipv4 host** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

**domain ipv4 host** *host-name v4address2.....v4address8*
**no domain ipv4 host** *host-name v4address1*

## Syntax Description

| | |
|---|---|
| host-name | Name of the host. The first character can be either a letter or a number. |
| v4address1 | Associated IP address. |
| v4address2...v4address8 | (Optional) Additional associated IP address. You can bind up to eight addresses to a hostname. |

## Command Default

No static mapping is configured.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The first character can be either a letter or a number. If you use a number, the operations you can perform (such as **ping**) are limited.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |
| basic-services | read, write |

## Examples

The following example shows how to define two IPv4 static mappings:

```
RP/0/RP0/CPU0:router(config)# domain ipv4 host host1 192.168.7.18
RP/0/RP0/CPU0:router(config)# domain ipv4 host bost2 10.2.0.2 192.168.7.33
```

# domain ipv6 host

To define a static hostname-to-address mapping in the host cache using IPv6, use the **domain ipv6 host** command in XR Config mode. To remove the **domain ipv6 host** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

**domain ipv6 host** *host-name v6address1 [v6address2 .....v6address4]*
**no domain ipv6 host** *host-name v6address1*

## Syntax Description

| | |
|---|---|
| host-name | Name of the host. The first character can be either a letter or a number. |
| v6address1 | Associated IP address. |
| v6address2...v6address4 | (Optional) Additional associated IP address. You can bind up to four addresses to a hostname. |

## Command Default

No static mapping is configured. IPv6 address prefixes are not enabled.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The first character can be either a letter or a number. If you use a number, the operations you can perform (such as **ping**) are limited.

## Task ID

| Task ID | Operations |
|---|---|
| ip services | read, write |

## Examples

The following example shows how to define two IPv6 static mappings:

```
RP/0/RP0/CPU0:router(config)# domain ipv6 host host1 ff02::2
RP/0/RP0/CPU0:router(config)# domain ipv6 host host2 ff02::1
```

# domain list

To define a list of default domain names to complete unqualified hostnames, use the **domain list** command in XR Config mode. To delete a name from a list, use the **no** form of this command.

**domain list** *domain-name*
**no domain list** *domain-name*

## Syntax Description

| | |
|---|---|
| domain-name | Domain name. Do not include the initial period that separates an unqualified name from the domain name. |

## Command Default

No domain names are defined.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If there is no domain list, the domain name that you specified with the domain name (IPAddr), on page 221 command is used to complete unqualified hostnames. If there is a domain list, the default domain name is not used. The **domain list** command is similar to the domain name (IPAddr), on page 221 command, except that you can use the **domain list** command to define a list of domains, each to be tried in turn.

## Task ID

| Task ID | Operations |
|---|---|
| ip-service | read, write |

## Examples

The following example shows how to add several domain names to a list:

```
RP/0/RP0/CPU0:router(config)# domain list domain1.com
RP/0/RP0/CPU0:router(config)# domain list domain2.edu
```

The following example shows how to add a name to and then delete a name from the list:

```
RP/0/RP0/CPU0:router(config)# domain list domain3.edu
RP/0/RP0/CPU0:router(config)# no domain list domain2.edu
```

## Related Commands

| Command | Description |
|---|---|
| domain name (IPAddr), on page 221 | Defines a default domain name to complete unqualified hostnames (names without a dotted-decimal domain name). |

| Command | Description |
| --- | --- |
| show hosts, on page 237 | Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses. |

# domain lookup disable

To disable the IP Domain Name System (DNS)-based hostname-to-address translation, use the **domain lookup disable** 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.

**domain lookup disable**
**no domain lookup disable**

## Syntax Description

This command has no keywords or arguments.

## Command Default

The IP DNS-based host-to-address translation is enabled.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Using the **no** command removes the specified command from the configuration file and restores the system to its default condition. The **no** form of this command is not stored in the configuration file.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to enable the IP DNS-based hostname-to-address translation:

```
RP/0/RP0/CPU0:router(config)# domain lookup disable
```

## Related Commands

| Command | Description |
|---|---|
| domain name (IPAddr), on page 221 | Defines a default domain name to complete unqualified hostnames (names without a dotted-decimal domain name). |
| domain name-server, on page 222 | Specifies the address of one or more name servers to use for name and address resolution. |
| show hosts, on page 237 | Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses. |

# domain name (IPAddr)

To define a default domain name that the software uses to complete unqualified hostnames, use the **domain name** command in the appropriate mode. To remove the name, use the **no** form of this command.

**domain name** *domain-name*
**no domain name** *domain-name*

## Syntax Description

| | |
|---|---|
| domain-name | Default domain name used to complete unqualified hostnames. Do not include the initial period that separates an unqualified name from the domain name. |

## Command Default

There is no default domain name.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## 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.

If a hostname does not contain a domain name, then a dot and the domain name configured by the **domain name** command are appended to the hostname before it is added to the host table.

If no domain name is configured by the **domain name** command and the user provides only the hostname, then the request is not looked up.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Related Commands

| Command | Description |
|---|---|
| domain list, on page 218 | Defines a list of default domain names to complete unqualified hostnames. |
| domain name-server, on page 222 | Specifies the address of one or more name servers to use for name and address resolution. |
| show hosts, on page 237 | Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses. |

# domain name-server

To specify the address of one or more name servers to use for name and address resolution, use the **domain name-server** command in XR Config mode. To remove the address specified, use the **no** form of this command.

**domain name-server** *server-address*
**no domain name-server** *server-address*

## Syntax Description

| | |
|---|---|
| server-address | IP address of a name server. |

## Command Default

If no name server address is specified, the default name server is 255.255.255.255. IPv4 and IPv6 address prefixes are not enabled.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

You can enter up to six addresses, but only one for each command.

If no name server address is specified, the default name server is 255.255.255.255 so that the DNS lookup can be broadcast to the local network segment. If a DNS server is in the local network, it replies. If not, there might be a server that knows how to forward the DNS request to the correct DNS server.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to specify host 192.168.1.111 as the primary name server and host 192.168.1.2 as the secondary server:

```
RP/0/RP0/CPU0:router(config)# domain name-server 192.168.1.111
RP/0/RP0/CPU0:router(config)# domain name-server 192.168.1.2
```

## Related Commands

| Command | Description |
|---|---|
| domain lookup disable, on page 220 | Disables the domain lookup. |
| domain name (IPAddr), on page 221 | Defines a default domain name to complete unqualified hostnames (names without a dotted-decimal domain name). |

# ftp client anonymous-password

To assign a password for anonymous users, use the **ftp client anonymous-password** command in XR Config mode. To remove the **ftp client anonymous-password** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

**ftp client anonymous-password** *password*
**no ftp client anonymous-password**

## Syntax Description

| | |
|---|---|
| password | Password for the anonymous user. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **ftp client anonymous-password** command is File Transfer Protocol (FTP) server dependent.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to set the anonymous password to *xxxx*:

```
RP/0/RP0/CPU0:router(config)# ftp client anonymous-password xxxx
```

## Related Commands

| Command | Description |
|---|---|
| ftp client passive, on page 224 | Configures the software to use only passive File Transfer Protocol (FTP) connections. |
| ftp client password, on page 225 | Specifies the password for the File Transfer Protocol (FTP) connections. |
| ftp client source-interface, on page 227 | Specifies the source IP address for File Transfer Protocol (FTP) connections. |
| ftp client username, on page 229 | Specifies the username for File Transfer Protocol (FTP) connections. |

# ftp client passive

To configure the software to use only passive File Transfer Protocol (FTP) connections, use the **ftp client passive** command in XR Config mode. To remove the **ftp client passive** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

**ftp client passive**
**no ftp client passive**

## Syntax Description

This command has no keywords or arguments.

## Command Default

FTP data connections are active.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Using the **ftp client passive** command allows you to make only passive-mode FTP connections. To specify the source IP address for FTP connections, use the **ftp client source-interface** command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to configure the networking device to use only passive FTP connections:

```
RP/0/RP0/CPU0:router(config)# ftp client passive

1d:3h:54:47: ftp_fs[16437]: FTP: verifying tuple passive (SET).
1d:3h:54:47: ftp_fs[16437]: FTP: applying tuple passive (SET).
1d:3h:54:47: ftp_fs[16437]: FTP: passive mode has been enabled.
```

## Related Commands

| Command | Description |
|---|---|
| ftp client anonymous-password, on page 223 | Assigns a password for anonymous users. |
| ftp client password, on page 225 | Specifies the password for the File Transfer Protocol (FTP) connections. |
| ftp client source-interface, on page 227 | Specifies the source IP address for File Transfer Protocol (FTP) connections. |
| ftp client username, on page 229 | Specifies the username for File Transfer Protocol (FTP) connections. |

# ftp client password

To specify the password for the File Transfer Protocol (FTP) connections, use the **ftp client password** command in XR Config mode. To disable this feature, use the **no** form of this command.

**ftp client password** {*clear-text-password* | **clear** *clear-text password* | **encrypted** *encrypted-text password*}
**no ftp client password** {*clear-text-password* | **clear** *clear-text password* | **encrypted** *encrypted-text password*}

## Syntax Description

| | |
|---|---|
| clear-text-password | Specifies an unencrypted (cleartext) user password |
| **clear** *clear-text password* | Specifies an unencrypted (cleartext) shared password. |
| **encrypted** *encrypted-text password* | Specifies an encrypted shared password. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to specify the password for the File Transfer Protocol (FTP) connections:

```
RP/0/RP0/CPU0:router(config)# ftp client password lab
```

## Related Commands

| Command | Description |
|---|---|
| ftp client anonymous-password, on page 223 | Assigns a password for anonymous users. |
| ftp client passive, on page 224 | Configures the software to use only passive File Transfer Protocol (FTP) connections. |
| ftp client source-interface, on page 227 | Specifies the source IP address for File Transfer Protocol (FTP) connections. |

| Command | Description |
| --- | --- |
| ftp client username, on page 229 | Specifies the username for File Transfer Protocol (FTP) connections. |

# ftp client source-interface

To specify the source IP address for File Transfer Protocol (FTP) connections, use the **ftp client source-interface** command in XR Config mode . To remove the **ftp client source-interface** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

**ftp client source-interface** *type interface-path-id*
**no ftp client source-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.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

The FTP source address is the IP address of the interface used by the FTP packets to leave the networking device.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use this command to set the same source address for all FTP connections. To configure the software to use only passive FTP connections, use the **ftp client passive** command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to configure the IP address associated with HundredGigEinterface 0/1/2/1 as the source address on all FTP packets, regardless of which interface is actually used to send the packet:

```
RP/0/RP0/CPU0:router(config)# ftp client source-interface HundredGigE0/1/2/1
```

## Related Commands

| Command | Description |
|---|---|
| ftp client anonymous-password, on page 223 | Assigns a password for anonymous users. |

| Command | Description |
|---|---|
| ftp client passive, on page 224 | Configures the software to use only passive File Transfer Protocol (FTP) connections. |
| ftp client password, on page 225 | Specifies the password for the File Transfer Protocol (FTP) connections. |
| ftp client username, on page 229 | Specifies the username for File Transfer Protocol (FTP) connections. |

# ftp client username

To specify the username for File Transfer Protocol (FTP) connections, use the **ftp client username** command in XR Config mode. To disable this feature, use the **no** form of this command.

**ftp client username** *username*
**no ftp client username** *username*

## Syntax Description

| | |
|---|---|
| username | Name for FTP user. |

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to specify the username for FTP connections:

```
RP/0/RP0/CPU0:router(config)# ftp client username brownfox
```

## Related Commands

| Command | Description |
|---|---|
| ftp client anonymous-password, on page 223 | Assigns a password for anonymous users. |
| ftp client passive, on page 224 | Configures the software to use only passive File Transfer Protocol (FTP) connections. |
| ftp client password, on page 225 | Specifies the password for the File Transfer Protocol (FTP) connections. |
| ftp client source-interface, on page 227 | Specifies the source IP address for File Transfer Protocol (FTP) connections |

# ping (network)

To check host reachability and network connectivity on IP networks, use the **ping** command in XR EXEC mode.

**ping** [{**ipv4** | **ipv6**}] [{*host-nameip-address*}] [**count** *number*] [**size** *number*] [**source** {*ip-addressinterface-name* | **type** *number*}] [**timeout** *seconds*] [**pattern** *number*] [**type** *number*] [**verbose**] [**donnotfrag**] [**validate**] [**sweep**]

## Syntax Description

| | |
|---|---|
| **ipv4** | (Optional) Specifies IPv4 address prefixes. |
| ipv6 | (Optional) Specifies IPv6 address prefixes. |
| host-name | (Optional) Hostname of the system to ping. |
| ip-address | (Optional) IP address of the system to ping. |
| **count** *number* | (Optional) Sets the repeat count. Range is 0 to 2147483647. |
| **size** *number* | (Optional) Sets the datagram size. Range is 36 to 18024 |
| source | (Optional) Identifies the source address or source interface. |
| **type** *number* | (Optional) Sets the type of service. Range is 0 to 255. Available when the **ipv4** keyword is specified. |
| **timeout** *seconds* | (Optional) Sets the timeout in seconds. Range is 0 to 3600. |
| **pattern** *number* | (Optional) Sets the data pattern. Range is 0 to 65535. |
| verbose | (Optional) Sets verbose output. |
| donnotfrag | (Optional) Sets the Don't Fragment (DF) bit in the IP header. |
| validate | (Optional) Validates the return packet. |
| sweep | (Optional) Sets the sweep ping. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The default value for the **ping** command refers only to the target IP address. No default value is available for the target IP address.

The ping program sends an echo request packet to an address and then waits for a reply. Ping output can help you evaluate path-to-host reliability, delays over the path, and whether the host can be reached or is functioning.

**Note** The **ping** (EXEC) command is supported only on IP networks.

If you enter the command without specifying either a hostname or an IP address, the system prompts you to specify the target IP address and several other command parameters. After specifying the target IP address, you can specify alternate values for the remaining parameters or accept the displayed default for each parameter.

If the system cannot map an address for a hostname, it returns an "%Unrecognized host or address, or protocol not running" error message.

To abnormally terminate a ping session, enter the escape sequence, which is, by default, Ctrl-C. Simultaneously press and release the Ctrl and C keys.

This table describes the test characters sent by the ping facility.

***Table 31: ping Test Characters***

| Character | Description |
|---|---|
| ! | Each exclamation point indicates receipt of a reply. |
| . | Each period indicates that the network server timed out while waiting for a reply. |
| ? | Unknown packet type. |
| U | A "destination unreachable" error protocol data unit (PDU) was received. |
| C | A "congestion experienced" packet was received. |
| M | Fragmentation is needed, but the "don't fragment" bit in the IP header is set. When this bit is set, the IP layer does not fragment the packet and returns an Internet Control Message Protocol (ICMP) error message to the source if the packet size is larger than the maximum transmission size. When this bit is not set, the IP layer fragments the packet to forward it to the next hop. |
| Q | A source quench packet was received. |

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write, execute |

## Examples

Although the precise dialog varies somewhat between IPv4 and IPv6, all are similar to the ping session, using default values shown in the following output:

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

Protocol [ipv4]:
Target IP address: 10.0.0.1
Repeat count [5]:
Datagram size [100]:
Timeout in seconds [2]:
Extended commands? [no]: yes
Source address or interface: 10.0.0.2
Type of service [0]:
```

```
Set DF bit in IP header? [no]:
Validate reply data? [no]: yes
Data pattern [0xABCD]:
Loose, Strict, Record, Timestamp, Verbose[none]:
Sweep range of sizes? [no]:
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.25.58.21, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 2/11/49 ms
```

If you enter a hostname or an address on the same line as the **ping** command, the command performs the default actions appropriate for the protocol type of that hostname or address, as shown in the following output:

```
RP/0/RP0/CPU0:router# ping server01

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.7.27, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/8/9 ms
```

# ping bulk (network)

To check reachability and network connectivity to multiple hosts on IP networks, use the **ping bulk** command in XR EXEC mode.

**ping bulk ipv4** [**input cli** [{**batch** | **inline**}]]

## Syntax Description

| | |
|---|---|
| **ipv4** | Specifies IPv4 address prefixes. |
| **input** | Specifies input mode. |
| **cli** | Specifies input via CLI. |
| **batch** | Pings after all destinations are input. |
| **inline** | Pings after each destination is input. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

You must hit the Enter button and then specify one destination address per line.

Maximum number of destinations you can specify in the cli or batch mode is 2000.

## Task ID

| Task ID | Operation |
|---|---|
| basic-services | read, write, execute |

## Related Commands

| Command | Description |
|---|---|
| ping (network), on page 230 | Checks host reachability and network connectivity on IP networks. |

# scp

To securely transfer a file from a local directory to a remote directory or from a remote directory to a local directory, use the **scp** command in XR EXEC mode.

**scp** {*local-directory username@location/directory*} */ filename* {*username@location/directory local-directory* } */ filename*

## Syntax Description

| | |
|---|---|
| *local-directory* | Specifies the local directory on the device. |
| *username@location/directory* | Specifies the remote directory where *location* is the IP address of the remote device. |
| *filename* | Specifies the file name to be transferred. |

## Command Default

None

## Usage Guidelines

Secure Copy Protocol (SCP) is a file transfer protocol which provides a secure and authenticated method for transferring files. SCP relies on SSHv2 to transfer files from a remote location to a local location or from local location to a remote location.

Use the **scp** command to copy a file from the local device to a destination device or from a destination device to the local device.

Using SCP, you can only transfer individual files. You cannot transfer a file from a remote device to another remote device.

SSH server process must be running on the remote device.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to copy a file using the **scp** command from a local directory to a remote directory:

```
RP/0/RP0/CPU0:router# scp /usr/file1.txt root@209.165.200.1:/root/file3.txt

Connecting to 209.165.200.1...
Password:
  Transferred 553065 Bytes
  553065 bytes copied in 0 sec (7576232)bytes/sec
```

The following example shows how to copy a file using the **scp** command from a remote directory to a local directory:

```
RP/0/RP0/CPU0:router# scp root@209.165.200.1:/root/file4.txt /usr/file.txt

Connecting to 209.165.200.1...
Password:
  Transferred 553065 Bytes
  553065 bytes copied in 0 sec (7576232)bytes/sec
```

# show cinetd services

To display the services whose processes are spawned by Cinetd when a request is received, use the **show cinetd services** command in XR EXEC mode.

**show cinetd services**

## Syntax Description

This command has no keywords or arguments.

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read |

## Examples

The following is sample is output from the **show cinetd services** command:

```
RP/0/RP0/CPU0:router# show cinetd services

Family Service   Proto   Port   ACL  max_cnt  curr_cnt wait  Program Option
================================================================================
v4     telnet    tcp      23 unlimited  0        nowait       telnet
v4     tftp      udp      69 unlimited  0        wait          tftpd   disk0
```

This table describes the significant fields shown in the display.

***Table 32: show cinetd services Command Field Descriptions***

| Field | Description |
|---|---|
| Family | Version of the network layer (IPv4 or IPv6). |
| Service | Network service (for example, FTP, Telnet, and so on). |
| Proto | Transport protocol used by the service (tcp or udp). |
| Port | Port number used by the service. |
| ACL | Access list used to limit the service from some hosts. |
| max_cnt | Maximum number of concurrent servers allowed for a service. |
| curr_cnt | Current number of concurrent servers for a service. |

| Field | Description |
|---|---|
| wait | Status of whether Cinetd has to wait for a service to finish before serving the next request. |
| Program | Name of the program for a service. |
| Option | Service-specific options. |

## Related Commands

| Command | Description |
|---|---|
| telnet server, on page 245 | Enables Telnet services on a networking device. |
| tftp server, on page 248 | Enables or disables the TFTP server or a feature running on the TFTP server. |

# show hosts

To display the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of hostnames and addresses, use the **show hosts** command in XR EXEC mode.

**show hosts** [*host-name*]

## Syntax Description

| | |
|---|---|
| host-name | (Optional) Name of the host about which to display information. If omitted, all entries in the local cache are displayed. |

## Command Default

Unicast address prefixes are the default when IPv4 address prefixes are configured.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read |

## Examples

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

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

Default domain is cisco.com
Name/address lookup uses domain service
Name servers are 255.255.255.255
Host              Flags        Age(hr)   Type       Address(es)
host1.cisco.com   (temp, OK)    1         IP          192.168.4.10
abc              (perm, OK)    0         IP         10.0.0.0 10.0.0.2 10.0.0.3
```

This table describes the significant fields shown in the display.

***Table 33: show hosts Command Field Descriptions***

| Field | Description |
|---|---|
| Default domain | Default domain used to complete the unqualified hostnames. |
| Name/address lookup | Lookup is disabled or uses domain services. |
| Name servers | List of configured name servers. |
| Host | Hostname. |

| Field | Description |
|---|---|
| Flags | Indicates the status of an entry.<br>• temp—Temporary entry entered by a name server; the software removes the entry after 72 hours of inactivity.<br>• perm—Permanent entry entered by a configuration command; does not time out.<br>• OK—Entry is believed to be valid.<br>• ??—Entry is considered suspect and subject to revalidation.<br>• EX—Entry has expired. |
| Age(hr) | Number of hours since the software most recently referred to the cache entry. |
| Type | Type of address (IPv4 or IPv6). |
| Address(es) | Address of the host. One host may have up to eight addresses. |

## Related Commands

| Command | Description |
|---|---|
| clear host, on page 215 | Deletes entries from the host-name-and-address cache. |
| domain list, on page 218 | Defines a list of default domain names to complete unqualified hostnames. |
| domain lookup disable, on page 220 | Disables the IP DNS-based hostname-to-address translation. |
| domain name (IPAddr), on page 221 | Defines a default domain name to complete unqualified hostnames (names without a dotted-decimal domain name). |
| domain name-server, on page 222 | Specifies the address of one or more name servers to use for name and address resolution. |

# telnet

To log in to a host that supports Telnet, use the **telnet** command in XR EXEC mode.

**telnet** {*ip-addresshost-name*} [*options*]

## Syntax Description

| | |
|---|---|
| ip-address | IP address of a specific host on a network.<br>• IPv4 address format—Must be entered in the (*x.x.x.x*) format.<br>• IPv6 address format— Must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| host-name | Name of a specific host on a network. |
| options | (Optional) Telnet connection options. See Table 34: Telnet Connection Options, on page 239for a list of supported options. |

## Command Default

Telnet client is in Telnet connection options nostream mode.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If the Telnet server is enabled, you should be able to start a Telnet session as long as you have a valid username and password.

This table lists the supported Telnet connection options.

***Table 34: Telnet Connection Options***

| Option | Description |
|---|---|
| /stream | Turns on stream processing, which enables a raw TCP stream with no Telnet control sequences. A stream connection does not process Telnet options and can be appropriate for connections to ports running UNIX-to-UNIX copy program (UUCP) and other non-Telnet protocols. |
| /nostream | Turns off stream processing. |

| Option | Description |
|---|---|
| port number | Port number. Range is 0 to 65535. |
| /source-interface | Specifies source interface. |

To display a list of the available hosts, use the **show hosts** command. To display the status of all TCP connections, use the **show tcp** command.

The software assigns a logical name to each connection, and several commands use these names to identify connections. The logical name is the same as the hostname, unless that name is already in use or you change the connection name with the **name-connection** EXEC command. If the name is already in use, the software assigns a null name to the connection.

The Telnet software supports special Telnet commands in the form of Telnet sequences that map generic terminal control functions to operating system-specific functions. To issue a special Telnet command, enter the escape sequence and then a command character. The default escape sequence is Ctrl-^ (press and hold the Control and Shift keys and the 6 key). You can enter the command character as you hold down Ctrl or with Ctrl released; you can use either uppercase or lowercase letters. Table 35: Special Telnet Escape Sequences, on page 240 lists the special Telnet escape sequences.

***Table 35: Special Telnet Escape Sequences***

| Escape Sequence[8] | Purpose |
|---|---|
| Ctrl-^ c | Interrupt Process (IP). |
| Ctrl-^ o | Terminate Output (AO). |
| Ctrl-^ u | Erase Line (EL). |

[8] The caret (^) symbol refers to Shift-6 on your keyboard.

At any time during an active Telnet session, you can list the Telnet commands by pressing the escape sequence keys followed by a question mark at the system prompt:

**ctrl-^?**

A sample of this list follows. In this sample output, the first caret (^) symbol represents the Control key, and the second caret represents Shift-6 on your keyboard:

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

[Special telnet escape help]
^^B  sends telnet BREAK
^^C  sends telnet IP
^^H  sends telnet EC
^^O  sends telnet AO
^^T  sends telnet AYT
^^U  sends telnet EL
```

You can have several concurrent Telnet sessions open and switch among them. To open a subsequent session, first suspend the current connection by pressing the escape sequence (Ctrl-Shift-6 and then x [Ctrl^x] by default) to return to the system command prompt. Then open a new connection with the **telnet** command.

To terminate an active Telnet session, issue any of the following commands at the prompt of the device to which you are connecting:

- close
- disconnect
- exit
- logout
- quit

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write, execute |

## Examples

The following example shows how to establish a Telnet session to a remote host named host1:

```
RP/0/RP0/CPU0:router# telnet host1
```

## Related Commands

| Command | Description |
|---|---|
| aaa authentication login default local | Sets AAA authentication at login. For more information, see *System Management Command Reference for Cisco NCS 6000 Series Routers*. |
| telnet server, on page 245 | Enables Telnet services on a networking device. |
| terminal length | Sets the number of lines on the current terminal screen for the current session. For more information, see *System Management Command Reference for Cisco NCS 6000 Series Routers*. |
| terminal width | Sets the number of character columns on the terminal screen for the current session. For more information, see *System Management Command Reference for Cisco NCS 6000 Series Routers*. |

# telnet client source-interface

To specify the source IP address for a Telnet connection, use the **telnet client source-interface** command in XR Config mode. To remove the **telnet client source-interface** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

**telnet** {**ipv4** | **ipv6**} **client source-interface** *type interface-path-id*
**no telnet client source-interface** *type interface-path-id*

## Syntax Description

| | |
|---|---|
| ipv4 | Specifies IPv4 address prefixes. |
| ipv6 | Specifies IPv6 address prefixes. |
| type | Interface type. For more information, use the question mark (?) online help function. |
| interface-path-id | Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

The IP address of the best route to the destination is used as the source IP address.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **telnet client source-interface** command to set the IP address of an interface as the source for all Telnet connections.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| ip-services | read, write |

## Examples

The following example shows how to set the IP address for HundredGigE interface 1/0/2/1 as the source address for Telnet connections:

```
RP/0/RP0/CPU0:router(config)# telnet ipv4 client source-interface HundredGigE1/0/2/1
```

## Related Commands

| Command | Description |
| --- | --- |
| telnet server, on page 245 | Enables Telnet services on a networking device. |

# telnet dscp

To define the differentiated services code point (DSCP) value and IPv4 precedence to specifically set the quality-of-service (QoS) marking for Telnet traffic on a networking device, use the **telnet dscp** command in XR Config mode. To disable DSCP, use the **no** form of this command.

**telnet ipv4 dscp** *dscp-value*
**no telnet ipv4 dscp** *dscp-value*

## Syntax Description

| | |
|---|---|
| ipv4 | Specifies IPv4 address prefixes. |
| dscp-value | Value for DSCP. The range is from 0 to 63. The default value is 0. |

## Command Default

If DSCP is disabled or not configured, the following default values are listed:

- The default value for the server 16.
- The default value for the client is 0.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

IPv4 is the supported protocol for defining a DSCP value for locally originated Telnet traffic.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| ip-services | read, write |

## Examples

The following example shows how to define the DSCP value and IPv4 precedence:

```
RP/0/RP0/CPU0:router(config)# telnet  ipv4 dscp 40
RP/0/RP0/CPU0:router(config)# telnet  ipv4 dscp 10
```

## Related Commands

| Command | Description |
|---|---|
| telnet, on page 239 | Logs in to a host that supports Telnet. |

# telnet server

To enable Telnet services on a networking device, use the **telnet server** command in XR Config mode. To disable Telnet services, use the **no** form of this command.

**telnet** {**ipv4** | **ipv6**} **server max-servers** *limit* [**access-list** *list-name*]
**no telnet** {**ipv4** | **ipv6**} **server max-servers** *limit* [**access-list** *list-name*]

## Syntax Description

| | |
|---|---|
| ipv4 | Specifies IPv4 address prefixes. |
| ipv6 | Specifies IPv6 address prefixes. |
| max-servers | Sets the number of allowable Telnet servers. |
| no-limit | Specifies that there is no maximum number of allowable Telnet servers. |
| limit | Specifies the maximum number of allowable Telnet servers. Range is 1 to 200. |
| **access-list** | (Optional) Specifies an access list. |
| *list-name* | (Optional) Access list name. |

## Command Default

Telnet services are disabled.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Disable Telnet services to prevent inbound Telnet connections from being accepted into a networking device using the **telnet** command. After Telnet services are disabled, no new inbound connections are accepted, and the Cisco Internet services daemon (Cinetd) stops listening on the Telnet port.

Enable Telnet services by setting the **max-servers** keyword to a value of one or greater. This allows inbound Telnet connections into a networking device.

This command affects only inbound Telnet connections to a networking device. Outgoing Telnet connections can be made regardless of whether Telnet services are enabled.

Using the **no** form of the command disables the connection and restores the system to its default condition.

**Note** Before establishing communications with the router through a Telnet session, configure the telnet server and vty-pool functions (see System Management Command Reference Guide, System Management Configuration Guide, and IP Addresses and Services Configuration Guide).

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| ip-services | read, write |

## Examples

The following example shows how to enable Telnet services for one server:

```
RP/0/RP0/CPU0:router(config)# telnet ipv4 server max-servers 1
```

## Related Commands

| Command | Description |
|---|---|
| telnet, on page 239 | Logs in to a host that supports Telnet. |

# tftp client source-interface

To specify the source IP address for a TFTP connection, use the **tftp client source-interface** command in XR Config mode. To remove the **tftp client source-interface** command from the configuration file and restore the system to its default condition, use the **no** form of this command.

**tftp client source-interface** *type interface-path-id*
**no tftp client source-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.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

The IP address of the best route to the destination is used as the source IP address.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **tftp client source-interface** command to set the IP address of an interface as the source for all TFTP connections.

## Task ID

| Task ID | Operations |
|---|---|
| ip-services | read, write |

## Examples

The following example shows how to set the IP address for HundredGigE interface 1/0/2/1 as the source address for TFTP connections:

```
RP/0/RP0/CPU0:router(config)# tftp client source-interface HundredGigE1/0/2/1
```

## Related Commands

| Command | Description |
|---|---|
| tftp server, on page 248 | Enables or disables the TFTP server or a feature running on the TFTP server. |

# tftp server

To enable or disable the TFTP server or a feature running on the TFTP server, use the **tftp server** command in XR Config mode. To restore the system to its default condition, use the **no** form of this command.

**tftp** {**ipv4** | **ipv6**} **server homedir** *tftp-home-directory* **max-servers** {*number* | **no-limit**} **access-list** *name*
**no tftp** {**ipv4** | **ipv6**} **server homedir** *tftp-home-directory* **max-servers** {*number* | **no-limit**} **access-list** *name*

## Syntax Description

| | |
|---|---|
| ipv4 | Specifies IPv4 address prefixes. |
| ipv6 | Specifies IPv6 address prefixes. |
| **homedir** *tftp-home-directory* | Specifies the home directory. |
| **max-servers** *number* | Sets the maximum number of concurrent TFTP servers. The range is from 1 to 2147483647. |
| **max-servers no-limit** | Sets no limit to process a number of allowable TFTP server. |
| **access-list** *name* | Specifies the name of the access list associated with the TFTP server. |

## Command Default

The TFTP server is disabled by default. When not specified, the default value for the **max-servers** keyword is unlimited.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Using the **no** form of the **tftp server** command removes the specified command from the configuration file and restores the system to its default condition. The **no** form of the command is not stored in the configuration file.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| ip-services | read, write |

## Examples

The following example shows that the TFTP server is enabled for the access list named test:

```
RP/0/RP0/CPU0:router(config)# tftp ipv4 server homedir disk0 access-list test max-servers
100
```

# traceroute

To discover the routes that packets actually take when traveling to their destination across an IP network, use the **traceroute** command in XR EXEC mode.

**traceroute** [{**ipv4** | **ipv6**}] [{*host-nameip-address*}] [**source** {*ip-addressinterface-name* }] [**numeric**] [**timeout** *seconds*] [**probe** *count*] [**minttl** *seconds*] [**maxttl** *seconds*] [**port** *number*] [**verbose**]

## Syntax Description

| | |
|---|---|
| **ipv4** | (Optional) Specifies IPv4 address prefixes. |
| ipv6 | (Optional) Specifies IPv6 address prefixes. |
| host-name | (Optional) Hostname of system to use as the destination of the trace attempt. |
| ip-address | (Optional) Address of system to use as the destination of the trace attempt. |
| source | (Optional) Source address. |
| *ip-address-name* | (Optional) IP address A.B.C.D or hostname. |
| numeric | (Optional) Numeric display only. |
| **timeout** *seconds* | (Optional) Timeout value. Range is 0 to 3600. |
| **probe** *count* | (Optional) Probe count. Range is 0 to 65535. |
| **minttl** *seconds* | (Optional) Minimum time to live. Range is 0 to 255. |
| **maxttl** *seconds* | (Optional) Maximum time to live. Range is 0 to 255. |
| **port** *number* | (Optional) Port number. Range is 0 to 65535. |
| **priority** *number* | (Optional) Packet priority. Range is 0 to 15. Available when the **ipv6** keyword is specified. |
| verbose | (Optional) Verbose output. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The default value for the **traceroute** command refers only to the destination. No default value is available for the destination address.

The **traceroute** command works by taking advantage of the error messages generated by networking devices when a datagram exceeds its time-to-live (TTL) value.

The **traceroute** command starts by sending probe datagrams with a TTL value of 1, which causes the first networking device to discard the probe datagram and send back an error message. The **traceroute** command sends several probes at each TTL level and displays the round-trip time for each.

The **traceroute** command sends out one probe at a time. Each outgoing packet may result in one or two error messages. A "time-exceeded" error message indicates that an intermediate networking device has seen and discarded the probe. A "destination-unreachable" error message indicates that the destination node has received the probe and discarded it because it could not deliver the packet. If the timer goes off before a response comes in, the **traceroute** command prints an asterisk (*).

The **traceroute** command terminates when the destination responds, when the maximum TTL is exceeded, or when the user interrupts the trace with the escape sequence, which is, by default, Ctrl-C. Simultaneously press and release the Ctrl and C keys.

To use nondefault parameters and invoke an extended **traceroute** test, enter the command without a *host-name* or *ip- address* argument. You are stepped through a dialog to select the desired parameter values for the **traceroute** test.

Because of how IP is implemented on various networking devices, the IP **traceroute** command may behave in unexpected ways.

Not all destinations respond correctly to a probe message by sending back an "ICMP port unreachable" message. A long sequence of TTL levels with only asterisks, terminating only when the maximum TTL has been reached, may indicate this problem.

There is a known problem with the way some hosts handle an "ICMP TTL exceeded" message. Some hosts generate an "ICMP" message, but they reuse the TTL of the incoming packet. Because this value is zero, the ICMP packets do not succeed in returning. When you trace the path to such a host, you may see a set of TTL values with asterisks (*). Eventually the TTL is raised high enough that the "ICMP" message can get back. For example, if the host is six hops away, the **traceroute** command times out on responses 6 through 11.

## Task ID

| Task ID | Operations |
|---|---|
| basic-services | read, write, execute |

## Examples

The following output shows a sample **traceroute** session when a destination hostname has been specified:

```
RP/0/RP0/CPU0:router# traceroute host8-sun

Type escape sequence to abort.
Tracing the route to 192.168.0.73
 1 192.168.1.6 (192.168.1.6) 10 msec 0 msec 10 msec
 2 gateway01-gw.gateway.cisco.com (192.168.16.2) 0 msec 10 msec 0 msec
 3 host8-sun.cisco.com (192.168.0.73) 10 msec * 0 msec
```

The following display shows a sample extended **traceroute** session when a destination hostname is not specified:

```
traceroute# traceroute

Protocol [ipv4]:
Target IP address: ena-view3
Source address or interface: 10.0.58.29
Numeric display? [no]:
Timeout in seconds [3]:
```

```
Probe count [3]:
Minimum Time to Live [1]:
Maximum Time to Live [30]:
Port Number [33434]:
Loose, Strict, Record, Timestamp, Verbose[none]:

Type escape sequence to abort.
Tracing the route to 171.71.164.199
 1  sjc-jpxlnock-vpn.cisco.com (10.25.0.1) 30 msec  4 msec  4 msec
 2  15lab-vlan725-gx1.cisco.com (173.19.72.2) 7 msec  5 msec  5 msec
 3  stc15-00lab-gw1.cisco.com (173.24.114.33) 5 msec  6 msec  6 msec
 4  stc5-lab4-gw1.cisco.com (173.24.114.89) 5 msec  5 msec  5 msec
 5  stc5-sbb4-gw1.cisco.com (172.71.241.162) 5 msec  6 msec  6 msec
 6  stc5-dc5-gw1.cisco.com (172.71.241.10) 6 msec  6 msec  5 msec
 7  stc5-dc1-gw1.cisco.com (172.71.243.2) 7 msec  8 msec  8 msec
 8  ena-view3.cisco.com (172.71.164.199) 6 msec  *  8 msec
```

This table describes the characters that can appear in traceroute output.

***Table 36: traceroute Text Characters***

| Character | Description |
|---|---|
| *xx* msec | For each node, the round-trip time in milliseconds for the specified number of probes. |
| * | Probe time out. |
| ? | Unknown packet type. |
| A | Administratively unreachable. This output usually indicates that an access list is blocking traffic. |
| H | Host unreachable. |
| N | Network unreachable. |
| P | Protocol unreachable. |
| Q | Source quench. |
| U | Port unreachable. |

# HSRP Commands

This chapter describes the Cisco IOS XR software commands used to configure and monitor the Hot Standby Router Protocol (HSRP).

For detailed information about HSRP concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# address (hsrp)

To enable hot standby protocol for IP, use the **address (hsrp)** command in the HSRP group submode. To disable hot standby protocol for IP, use the **no** form of this command.

**address** {**learn***address*}
**no address** {**learn***address*}

## Syntax Description

| | |
|---|---|
| **learn** | Learns virtual IP address from peer. |
| *address* | Hot standby IP address. |

## Command Default

None

## Command Modes

HSRP Group Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read, write |

### Example

This example shows how to enable a group to learn the primary virtual IPv4 address from received HSRP control packets:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 1 version 2
RP/0/RP0/CPU0:router(config-hsrp-gp)# address learn
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

## Related Commands

| Command | Description |
|---|---|
| address secondary (hsrp), on page 260 | Configures the secondary virtual IPv4 address for a virtual router. |

# address global (HSRP)

To configure the global virtual IPv6 address for the HSRP group, use the **address global** command in the virtual router submode. To deconfigure the global virtual IPv6 address for the HSRP group, use the **no** form of this command.

**address global** *ipv6-address*

**no address global** *ipv6-address*

**Syntax Description**

| | |
|---|---|
| *ipv6-address* | Global HSRP IPv6 address. |

**Command Default**

None

**Command Modes**

HSRP Group Submode, under the IPv6 address-family

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operation |
|---|---|
| hsrp | read,write |

### Example

This example shows how to add a global virtual IPv6 address for the HSRP group:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface tenGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv6
RP/0/RP0/CPU0:router(config-hsrp-address-family)# hsrp 3
RP/0/RP0/CPU0:router(config-hsrp-virtual-router)# address global 4000::1000
RP/0/RP0/CPU0:router(config-hsrp-virtual-router)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

# address global subordinate (HSRP)

To configure the global virtual IPv6 address for the subordinate group, use the **address global** command in the HSRP slave submode. To deconfigure the global virtual IPv6 address for the subordinate group, use the **no** form of this command.

**address global** *ipv6-address*

**no address global** *ipv6-address*

**Syntax Description**

| | |
|---|---|
| *ipv6-address* | Global VRRP IPv6 address. |

**Command Default**

None

**Command Modes**

HSRP Slave Submode, under the IPv6 address-family

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operation |
|---|---|
| hsrp | read,write |

### Example

This example shows how to add a global virtual IPv6 address for the subordinate group:

```
Router# configure
Router(config)# router hsrp
Router(config-hsrp)# interface tenGigE 0/4/0/4
Router(config-hsrp-if)# address-family ipv6
Router(config-hsrp-address-family)# hsrp 3 slave
Router(config-hsrp-virtual-router)# address global 4000::1000
Router(config-hsrp-virtual-router)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

# address linklocal (HSRP)

To either configure the virtual link-local IPv6 address for the subordinate group or to specify that the virtual link-local IPv6 address should be enabled and calculated automatically from the virtual router virtual Media Access Control (MAC) address, use the **address linklocal** command in the virtual router submode. To deconfigure the virtual link-local IPv6 address for the subordinate group, use the **no** form of this command.

**address linklocal**
*ipv6-address* | **autoconfig**

**no address linklocal**
*ipv6-address* | **autoconfig**

## Syntax Description

| | |
|---|---|
| *ipv6-address* | HSRP IPv6 link-local address. |
| **autoconfig** | Autoconfigures the HSRP IPv6 link-local address. |

## Command Default

None

## Command Modes

HSRP Slave Submode, under the IPv6 address-family

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

When you configure HSRP for IPv6, you must also configure the linklocal IPv6 address using either the *ipv6-address* argument or the **autoconfig** keyword. If you configure only the global IPv6 address and commit the changes using the **commit** keyword, the router does not accept the configuration and displays an error message.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read, write |

### Example

This example shows how to autoconfigure the HSRP IPv6 link-local address:

```
Router#configure
Router(config)#router hsrp
Router(config-hsrp)#interface tenGigE 0/4/0/4
Router(config-hsrp-if)#address-family ipv6
Router(config-hsrp-address-family)#hsrp 3 slave
Router(config-hsrp-virtual-router)#address linklocal autoconfig
Router(config-hsrp-virtual-router)#
```

This example shows how to configure the virtual link-local IPv6 address for the subordinate group:

```
Router#configure
Router(config)#router hsrp
Router(config-hsrp)#interface tenGigE 0/4/0/4
Router(config-hsrp-if)#address-family ipv6
Router(config-hsrp-address-family)#hsrp 3 slave
Router(config-hsrp-virtual-router)#address linklocal FE80::260:3EFF:FE11:6770
Router(config-hsrp-virtual-router)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

# address secondary (hsrp)

To configure the secondary virtual IPv4 address for a virtual router, use the **address secondary** command in the Hot Standby Router Protocol (HSRP) virtual router submode. To deconfigure the secondary virtual IPv4 address for a virtual router, use the **no** form of this command.

**address** *address* **secondary**
**no address** *address* **secondary**

## Syntax Description

| | |
|---|---|
| **secondary** | Sets the secondary HSRP IP address. |
| *address* | HSRP IPv4 address. |

## Command Default

None

## Command Modes

HSRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read, write |

### Example

This example shows how to set the secondary virtual IPv4 address for the virtual router:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface tenGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 3 version 2
RP/0/RP0/CPU0:router(config-hsrp-gp)# address 10.20.30.1 secondary
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

## Related Commands

| Command | Description |
|---|---|
| address (hsrp), on page 254 | Enables hot standby protocol for IP. |

# authentication (hsrp)

To configure an authentication string for the Hot Standby Router Protocol (HSRP), use the **hsrp authentication** command in HSRP group submode. To delete an authentication string, use the **no** form of this command.

**authentication** *string*
**no authentication** [*string*]

## Syntax Description

| | |
|---|---|
| string | Authentication string. It can be up to eight characters long. The default is 'cisco'. |

## Command Default

The default authentication string is cisco.

## Command Modes

HSRP Group Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The authentication string is sent unencrypted in all HSRP messages. The same authentication string must be configured on all routers and access servers on a LAN to ensure interoperation. Authentication mismatch prevents a device from learning the designated Hot Standby IP address and the Hot Standby timer values from other routers configured with HSRP.

The **hsrp authentication** command is available for version 1 groups only

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

This example shows how to configure “company1” as the authentication string required to allow Hot Standby routers in group 1 on HundredGigE interface 0/4/0/4 to interoperate:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 1 version 1
RP/0/RP0/CPU0:router(config-hsrp-gp)# authentication company1
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

**Note** The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# bfd fast-detect (hsrp)

To enable bidirectional forwarding(BFD) fast-detection on a HSRP interface, use the **hsrp bfd fast-detect** command in HSRP group submode. This creates a BFD session between the HSRP router and its peer, and if the session goes down while HSRP is in backup state, this will initiate a HSRP failover. To disable BFD fast-detection, use the **no** form of this command.

**bfd fast-detect** [ **peer ipv4** *ipv4-address* [ *interface-type interface-path-id* ] ]

## Syntax Description

| | |
|---|---|
| **peer ipv4** *ipv4-address* | (Optional) BFD peer interface IPv4 address. |
| *interface-type interface-path-id* | (Optional) Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

BFD is disabled.

## Command Modes

HSRP Group Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

This example shows how to enable bfd fast-detect:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 1 version 2
RP/0/RP0/CPU0:router(config-hsrp-gp)# bfd fast-detect
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

## Related Commands

| Command | Description |
|---|---|
| hsrp bfd multiplier, on page 268 | Configures the multiplier value for BFD. |
| hsrp bfd minimum-interval, on page 267 | Configures the BFD minimum interval to be used for all HSRP BFD sessions on a given interface |

# clear hsrp statistics

To reset the Hot Standby Routing Protocol Statistics (HSRP) statistics to zero, use the **clear hsrp statistics** command in XR EXEC mode.

**clear hsrp statistics** [**interface** *interface-type interface-path-id* [*group*]]

## Syntax Description

| | |
|---|---|
| **interface** *interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| *group* | (Optional) Group number. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read, write |

### Example

This sample output is from the **clear hsrp statistics** command:

```
RP/0/RP0/CPU0:router# clear hsrp statistics
```

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# hsrp bfd minimum-interval

To configure the BFD minimum interval to be used for all HSRP BFD sessions on a given interface, use the **hsrp bfd minimum-interval** command in the interface configuration mode. To remove the configured minimum-interval period and set the minimum-interval period to the default period, use the **no** form of this command.

**hsrp bfd minimum-interval** *interval*
**no hsrp bfd minimum-interval** *interval*

## Syntax Description

| | |
|---|---|
| interval | Specify the minimum-interval in milliseconds. Range is 15 to 30000. |

## Command Default

Default minimum interval is 15 ms.

## Command Modes

HSRP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Minimum interval determines the frequency of sending BFD packets to BFD peers. It is the time between successive BFD packets sent for the session. Minimum interval is defined in milliseconds. The configured minimum interval applies to all BFD sessions on the interface.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

The following example shows how to configure a minimum interval of 100 milliseconds:

```
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface gig 0/1/1/0
RP/0/RP0/CPU0:router(config-hsrp-if)# hsrp bfd minimum-interval 100
```

## Related Commands

| Command | Description |
|---|---|
| hsrp bfd fast-detect | Enables BFD fast-detection on a HSRP interface. |
| hsrp bfd multiplier, on page 268 | Configures the multiplier value for BFD. |

# hsrp bfd multiplier

To set the BFD multiplier value, use the **hsrp bfd multiplier** command in the interface configuration mode. To remove the configured multiplier value and set the multiplier to the default value, use the **no** form of this command.

**hsrp bfd multiplier** *multiplier*
**no hsrp bfd multiplier** *multiplier*

## Syntax Description

| | |
|---|---|
| multiplier | Specifies the BFD multiplier value. Range is 2 to 50. |

## Command Default

Default value is 3.

## Command Modes

HSRP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The multiplier value specifies the number of consecutive BFD packets that, if not received as expected, cause a BFD session to go down. The BFD multiplier applies to all configured BFD sessions on the interface.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

The following example shows how to configure a BFD multiplier with multiplier value of 10:

```
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface gig 0/1/1/0
RP/0/RP0/CPU0:router(config-hsrp-if)# hsrp bfd multiplier 10
```

## Related Commands

| Command | Description |
|---|---|
| hsrp bfd fast-detect | Enables BFD fast-detection on a HSRP interface. |

# hsrp delay

To configure the activation delay for the Hot Standby Router Protocol (HSRP), use the **hsrp delay** command in HSRP interface configuration mode. To delete the activation delay, use the **no** form of this command.

**hsrp delay minimum** *value* **reload** *value*
**no hsrp delay**

## Syntax Description

| | |
|---|---|
| **minimum** *value* | Sets the minimum delay in seconds for every interface up event. Range is 0 to 10000. |
| **reload** *value* | Sets the reload delay in seconds for first interface up event. Range is 0 to 10000. |

## Command Default

**minimum** *value* : 1

**reload** *value* : 5

## Command Modes

HSRP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **hsrp delay** command delays the start of the HSRP finite state machine (FSM) on an interface up event to ensure that the interface is ready to pass traffic. This ensures that there are no mistaken state changes due to loss of hello packets. The minimum delay is applied on all interface up events and the reload delay is applied on the first interface event.

The values of zero must be explicitly configured to turn this feature off.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

The following example shows how to configure a minimum delay of 10 seconds with a reload delay of 100 seconds:

```
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface  0/RP0/CPU0/0
RP/0/RP0/CPU0:router(config-hsrp-if)# hsrp delay minimum 10 reload 100
```

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# hsrp ipv4

To activate the Hot Standby Router Protocol (HSRP), use the **hsrp ipv4** command in HSRP interface configuration mode. To disable HSRP, use the **no** form of this command.

**hsrp ipv4** [*ip-address* [**secondary**]]
**no hsrp ipv4** [*ip-address* [**secondary**]]

## Syntax Description

| | |
|---|---|
| group-number | (Optional) Group number on the interface for which HSRP is being activated. Range is 0 to 255. Default is 0. |
| ip-address | (Optional) IP address of the Hot Standby router interface. |
| secondary | (Optional) Indicates that the IP address is a secondary Hot Standby router interface. Useful on interfaces with primary and secondary addresses; you can configure primary and secondary HSRP addresses. |

## Command Default

*group-number* : 0

HSRP is disabled by default.

## Command Modes

HSRP interface configuration

## Usage Guidelines

The **hsrp ipv4** command activates HSRP on the configured interface. If an IP address is specified, that address is used as the designated address for the Hot Standby group. If no IP address is specified, the virtual address is learned from the active router. For HSRP to elect a designated router, at least one router in the Hot Standby group must have been configured with, or must have learned, the designated address. Configuring the designated address on the active router always overrides a designated address that is currently in use.

When the **hsrp ipv4** command is enabled on an interface, the handling of proxy Address Resolution Protocol (ARP) requests is changed (unless proxy ARP was disabled). If the Hot Standby state group has been configured with or has learned the designated address, the proxy ARP requests are answered using the MAC address of the Hot Standby group. Otherwise, proxy ARP responses are suppressed.

Configuring secondary Hot Standby router IP addresses is necessary when the interface has secondary IP addresses configured and redundancy must be provided for the networks of these addresses also.

A primary address must be configured before a secondary address. Likewise, a secondary address must be unconfigured before unconfiguring a primary address. All IP addresses can be unconfigured using the **no hsrp ipv4** command.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

The following example shows how to activate HSRP for group 1 on HundredGigE interface 0/2/0/1. The IP address used by the Hot Standby group is learned using HSRP.

```
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:routerrouter(config-hsrp)# interface HundredGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-hsrp-if)# hsrp 1 ipv4
```

## Related Commands

| Command | Description |
|---|---|
| hsrp redirects, on page 272 | Configures ICMP redirect messages to be sent when the HSRP is configured on an interface. |
| show hsrp, on page 281 | Displays HSRP information. |

# hsrp redirects

To configure Internet Control Message Protocol (ICMP) redirect messages to be sent when the Hot Standby Router Protocol (HSRP) is configured on an interface, use the **hsrp redirects** command in HSRP interface configuration mode. To revert to the default, which is that ICMP messages are enabled, use the **no** form of this command.

**hsrp redirects disable**
**no hsrp redirects disable**

## Syntax Description

| | |
|---|---|
| disable | Disables the filtering of ICMP redirect messages on interfaces configured with HSRP. |

## Command Default

HSRP ICMP redirects are enabled by default.

## Command Modes

HSRP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **hsrp redirects** command can be configured on a per-interface basis. When HSRP is first configured on an interface, the setting for that interface inherits the global value. With the **hsrp redirects** command is enabled, ICMP redirects messages are filtered by replacing the real IP address in the next-hop address of the redirect packet with a virtual IP address if it is known to HSRP.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

The following example shows how to allow HSRP to filter redirect messages on HundredGigE interface 0/2/0/1:

```
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/2/0/1

RP/0/RP0/CPU0:router(config-hsrp-if)# hsrp redirects disable
```

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# hsrp use-bia

To configure the Hot Standby Router Protocol (HSRP) to use the burned-in address of the interface as its virtual MAC address, instead of the preassigned MAC address or the functional address, use the **hsrp use-bia** command in HSRP interface configuration mode. To restore the default virtual MAC address, use the **no** form of this command.

**hsrp use-bia**
**no hsrp use-bia**

## Command Default

HSRP uses the preassigned MAC address on Ethernet.

## Command Modes

HSRP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

It is desirable to configure the **hsrp use-bia** command on an interface if there are devices that reject Address Resolution Protocol (ARP) replies with source hardware addresses set to a functional address.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

In the following example, the burned-in address of HundredGigE interface 0/2/0/1 will be the virtual MAC address mapped to the virtual IP address for all Hot Standby groups configured on HundredGigE interface 0/1/0/1:

```
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-hsrp-if)# hsrp use-bia
```

## Related Commands

| Command | Description |
|---|---|
| hsrp mac-address | Specifies a virtual MAC address for HSRP. |
| show hsrp, on page 281 | Displays HSRP information. |

# interface (HSRP)

To enable Hot Standby Router Protocol (HSRP) interface configuration command mode, use the **interface** command in router configuration mode. To terminate interface mode, 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.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

HSRP is disabled.

## Command Modes

Router HSRP configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

All the commands used to configure HSRP are used in HSRP interface configuration mode.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

The following example show how to enable HSRP interface configuration mode on HundredGigE 0/2/0/1:

```
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-hsrp-if)#
```

## Related Commands

| Command | Description |
|---|---|
| router hsrp, on page 279 | Enables HSRP. |

# preempt (hsrp)

To configure Hot Standby Router Protocol (HSRP) preemption and preemption delay, use the **hsrp preempt** command in HSRP group submode. To restore the default values, use the **no** form of this command.

**preempt** [**delay** *seconds*]
**no preempt** [**delay** *seconds*]

## Syntax Description

| | |
|---|---|
| **delay** *seconds* | (Optional) Time in seconds. The *seconds* argument causes the local router to postpone the taking over the active role for the specified preempt delay *seconds* value. Range is from 0 to 3600 (1 hour). Default is 0 (no delay). |

## Command Default

The default delay is 0.

## Command Modes

HSRP Group Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

When the **hsrp preempt** command is configured, the local router should attempt to assume control as the active router, if it has a hot standby priority higher than the current active router. If the hsrp preempt command is not configured, the local router assumes control as the active router only if no other router is currently in the active state.

When a router first comes up, it does not have a complete routing table. If HSRP is configured to preempt, the local HSRP group may become the active router, yet it is unable to provide adequate routing services. This problem can be solved by configuring a delay before the preempting router actually preempts the currently active router.

The preempt delay *seconds* value does not apply if there is no router currently in the active state. In this case, the local router becomes active after the appropriate timeouts (see the **hsrp timers** command), regardless of the preempt *delay seconds* value.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

This example, the router waits for 300 seconds (5 minutes) after having determined that it should preempt before attempting to preempt the active router. The router might become the active router in a shorter span of time despite the configured delay, if no active router is present. Only preempting the active router is delayed.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
```

```
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 1 version 2
RP/0/RP0/CPU0:router(config-hsrp-gp)# preempt delay 300
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

## Related Commands

| Command | Description |
|---|---|
| hsrp priority | Configures HSRP priority. |
| hsrp track | Configures an interface so that the Hot Standby priority changes based on the availability of other interfaces. |
| show hsrp, on page 281 | Displays HSRP information. |

# priority (hsrp)

To configure Hot Standby Router Protocol (HSRP) priority, use the **priority** command in HSRP group submode. To restore the default values, use the **no** form of this command.

**priority** *priority*
**no priority** *priority*

## Syntax Description

| | |
|---|---|
| *priority* | Priority value that prioritizes a potential Hot Standby router. Range is from 1 to 255. Default is 100. |

## Command Default

The default priority is 100.

## Command Modes

HSRP group submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The assigned priority is used to help select the active and standby routers. Assuming that preemption is enabled, the router with the highest priority becomes the designated active router. In case of ties, the interface IP addresses are compared, and the interface with the higher IP address has priority.

The priority of the device can change dynamically if an interface is configured with the **hsrp track** command and another interface on the device goes down.

If preemption is not enabled, the router may not become active even though it might have a higher priority than other HSRP routers.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

In this example, the router has a priority of 120:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 1 version 2
RP/0/RP0/CPU0:router(config-hsrp-gp)# priority 120
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

## Related Commands

| Command | Description |
|---|---|
| hsrp preempt | Configures HSRP preemption and preemption delay. |
| hsrp track | Configures an interface so that the Hot Standby priority changes based on the availability of other interfaces. |
| show hsrp, on page 281 | Displays HSRP information. |

# router hsrp

To enable the Hot Standby Router Protocol (HSRP), use the **router hsrp** command in XR Config mode. To disable HSRP, use the **no** form of this command.

**router hsrp**
**no router hsrp**

## Syntax Description

This command has no keywords or arguments.

## Command Default

HSRP is disabled.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

HSRP configuration commands must be configured in the HSRP interface configuration mode.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

The following example shows how to configure an HSRP redundancy process that contains a virtual router group 1 on HundredGigE 0/2/0/1:

```
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/2/0/1
RP/0/RP0/CPU0:router(config-hsrp-if)# hsrp 1 priority 254
```

# session name

To configure an HSRP session name, use the **session name** command in the HSRP group submode. To deconfigure an HSRP session name, use the **no** form of this command.

**name** *name*

**Syntax Description**

| | |
|---|---|
| *name* | MGO session name |

**Command Default**

None

**Command Modes**

HSRP Group Submode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operation |
|---|---|
| hsrp | read |

### Example

This example shows how to configure an HSRP session name.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface tenGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 1 version 2
RP/0/RP0/CPU0:router(config-hsrp-gp)# name s1
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

**Related Commands**

| Command | Description |
|---|---|
| hsrp mac-address | Configures a virtual MAC address for the Hot Standby Router Protocol (HSRP). |

# show hsrp

To display Hot Standby Router Protocol (HSRP) information, use the **show hsrp** command in XR EXEC mode mode.

**show hsrp** [*interface-type interface-path-id*] [*group-number*] [{**brief** | **detail**}]

## Syntax Description

| | |
|---|---|
| *interface-type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| *group-number* | (Optional) Group number on the interface for which output is displayed. |
| **brief** | (Optional) A single line of output summarizes each standby group. The **brief** keyword is the default if **detail** is not specified. |
| **detail** | (Optional) This keyword has the same effect as not specifying **brief**; more output is provided. |
| | (Optional) After this vertical bar (\|), specify one of these output modifiers and a keyword from the output:<br>• **begin** —Begins the output from the word that you specify.<br>• **exclude** —Excludes lines that match the word that you specify.<br>• **include** —Includes lines that match the word that you specify. |

## Command Default

By default, a single line of output summarizing each standby group is displayed.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **show hsrp** command to display HSRP information.

If you want to specify a value for the *group-number* argument, you must also specify an interface *type* and *number*.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read |

## Examples

This is sample output from the **show hsrp detail** command:

```
RP/0/RP0/CPU0:router# show hsrp detail

HundredGigE 0/4/0/0 - Group 1
  Local state is Active, priority 100
  Hellotime 3 sec holdtime 10 sec
  Next hello sent in 0.539
  Minimum delay 1 sec, reload delay 5 sec
BFD enabled: state none, interval 15 ms multiplier 3
  Hot standby IP address is 4.0.0.100 configured
  Active router is local
  Standby router is unknown expired
  Standby virtual mac address is 0000.0c07.ac01
  2 state changes, last state change 00:05:20
```

This table describes the significant fields shown in the display.

***Table 37: show hsrp Command Field Descriptions***

| Field | Description |
|---|---|
| HundredGigEE0/2/0/4 | Interface type and number and Hot Standby group number for the interface. |
| Local state is | State of local networking device; can be one of the following:<br>• Active—Current Hot Standby router.<br>• Standby—Router next in line to be the Hot Standby router.<br>• Speak—Router is sending packets to claim the active or standby role.<br>• Listen—Router is neither active nor standby, but if no messages are received from the active or standby router, it will start to “speak.”<br>• Learn—Router is neither active nor standby, nor does it have enough information to attempt to claim the active or standby roles.<br>• Init—Router is not yet ready to participate in HSRP, possibly because the associated interface is not up. |
| Hellotime | Current time (in seconds) between sending of hello packets, learned dynamically from the hello packets received from the active Hot Standby router. |
| holdtime | Current time (in seconds) before other routers declare the active or standby router to be down, learned dynamically from the hello packets received from the active Hot Standby router. |
| Next hello sent in | Time in which the software will send the next hello packet (in hours:minutes:seconds). |
| BFD enabled | Displays BFD related information (with multiplier and minimum interval details) |

| Field | Description |
|---|---|
| Hot standby IP address is configured | IP address of the current Hot Standby router. The word "configured" indicates that this address is known through the **hsrp ip** command. Otherwise, the address was learned dynamically through HSRP hello packets from other routers that do have the HSRP IP address configured. |
| Active router is | Value can be "local" or an IP address. Address of the current active Hot Standby router. |
| Standby router is | Value can be "local" or an IP address of the standby router (the router that is next in line to be the Hot Standby router). |
| Standby virtual mac address is | MAC address associated with the standby group address. |
| state changes | Number of times the router changed the standby state. |
| last state change | Time (in hours:minutes:seconds) expired since the last state change. |
| Tracking interface states for | List of interfaces that are being tracked and their corresponding states. Based on the **hsrp track** command. |
| Priority decrement | Value by which the standby priority is decremented or incremented when the tracked interface goes down or up, respectively. Default is 10. |

## Related Commands

| Command | Description |
|---|---|
| hsrp authentication | Configures an authentication string for HSRP. |
| hsrp ipv4 | Activates the HSRP. |
| hsrp mac-address | Specifies a virtual MAC address for HSRP. |
| hsrp preempt | Configures HSRP preemption and preemption delay. |
| hsrp priority | Configures HSRP priority. |
| hsrp timers | Configures the time between hello packets and the time before other routers declare the active Hot Standby or standby router to be down. |
| hsrp track | Configures an interface so that the Hot Standby priority changes based on the availability of other interfaces. |
| hsrp use-bia, on page 273 | Configures HSRP to use the burned-in address of the interface as its virtual MAC address, instead of the preassigned MAC address. |

# show hsrp bfd

To display Hot Standby Router Protocol (HSRP) bfd information across all interfaces, use the **show hsrp bfd** command in XR EXEC mode mode.

**show hsrp bfd** [*interface-type interface-path-id* [*ip-address*]]

## Syntax Description

| | |
|---|---|
| *interface-type*<br>*interface-path-id* | (Optional) Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| *ip-address* | (Optional) Destination IP address for BFD session. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read |

### Example

This example shows Hot Standby Router Protocol (HSRP) bfd information across all interfaces.

```
RP/0/RP0/CPU0:router# show hsrp bfd

BFD Interface    Destination IP  State    Intv Mult HSRP Interface   Grp
-------------    --------------  -----    ---- ---- --------------   ---
Gi0/3/0/2        10.0.0.2              up  100    3 Gi0/3/0/2          1
                                                    Gi0/3/0/2          2
Gi0/3/0/2        10.0.0.3        inactive  100    3 Gi0/3/0/2          3
                                                    Gi0/3/0/2          6
Gi0/3/0/3.1      10.0.1.2             down   15    3 Gi0/3/0/2          4
```

This example shows Hot Standby Router Protocol (HSRP) bfd information for the HundredGigE 0/3/0/2 interface.

```
RP/0/RP0/CPU0:router# show hsrp bfd HundredGigE 0/3/0/2 10.0.0.2

BFD Interface    Destination IP  State    Intv Mult HSRP Interface   Grp
-------------    --------------  -----    ---- ---- --------------   ---
Gi0/3/0/2        10.0.0.2              up  100    3 Gi0/3/0/2          1
                                                    Gi0/3/0/2          2
```

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# show hsrp mgo

To display Hot Standby Router Protocol (HSRP) mgo information across all interfaces, use the **show hsrp mgo** command in XR EXEC mode.

**show hsrp mgo** [{**brief** *session-name*}]

**Syntax Description**

| | |
|---|---|
| **brief** | (Optional) Displays information in a brief format. |
| *session-name* | (Optional) Display information for a single MGO Session. |

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operation |
|---|---|
| hsrp | read |

### Example

This example shows Hot Standby Router Protocol (HSRP) mgo information for interface HSRP3.

```
RP/0/RP0/CPU0:router# show hsrp mgo HSRP3

HSRP3
  Primary group Bundle-Ether1.1 IPv4 group 1
  State is Active
  Slave groups:
    Interface                Grp
    Bundle-Ether1.2            2
    Bundle-Ether1.3            3
    Bundle-Ether1.4            4
    Bundle-Ether1.5            5
```

This example shows Hot Standby Router Protocol (HSRP) mgo information across all interfaces in a brief format.

```
RP/0/RP0/CPU0:router# show hsrp mgo brief

Name            Interface      AF   Grp    State Slaves
```

```
HSRP1           Gi0/0/0/1       IPv4   1   Active     100
HSRP2           Te0/1/0/0.1     IPv4   2  Standby      50
HSRP3           BE1             IPv4   1   Active       4
HSRP4           BE1             IPv6  10   Active      11
```

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# show hsrp statistics

To display Hot Standby Router Protocol (HSRP) statistics information across all interfaces, use the **show hsrp statistics** command in XR EXEC mode.

**show hsrp** [{*interface-type* *interface-path-id*}] **statistics**

## Syntax Description

| | |
|---|---|
| *interface-type*<br>*interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read |

### Example

This sample output is from the **show hsrp statistics** command:

```
 RP/0/RP0/CPU0:router# show hsrp statistics
Protocol:
  Transitions to Active                2
  Transitions to Standby               2
  Transitions to Speak                 0
  Transitions to Listen                2
  Transitions to Learn                 0
  Transitions to Init                  0

Packets Sent:                         12
  Hello:                               7
  Resign:                              0
  Coup:                                2
  Adver:                               3

Valid Packets Received:                13
  Hello:                               8
  Resign:                              2
  Coup:                                0
```

```
  Adver:                                3

Invalid packets received:               0
  Too long:                             0
  Too short:                            0
  Mismatching/unsupported versions:     0
  Invalid opcode:                       0
  Unknown group:                        0
  Inoperational group:                  0
  Conflicting Source IP:                0
  Failed Authentication:                2
  Invalid Hello Time:                   0
  Mismatching Virtual IP:               0
```

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# show hsrp summary

To display Hot Standby Router Protocol (HSRP) summary information across all interfaces, use the **show hsrp summary** command in XR EXEC mode mode.

**show hsrp summary**

## Syntax Description

This command has no keywords or arguments.

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read |

### Example

This sample output is from the **show hsrp summary** command:

```
RP/0/RP0/CPU0:router# show hsrp summary
                Groups                VIPs
State  Sessions Slaves Total     Up  Down  Total
-----   ---------------------   -----------------
ALL          60    900    960    860  2020   2880

ACTIVE       10    190    200    200   300    500
STANDBY      15    235    250    250   600    850
SPEAK        10    190    200    200   400    600
LISTEN       10    190    200    200   400    600
LEARN         5      5     10     10    20     30
INIT         10     90    100      0   300    300

48   HSRP IPv4 interfaces    (43   up, 5   down)
5   Tracked IPv4 interfaces (4    up, 1    down)
5    BFD sessions       (3    up, 2    down)
```

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# hsrp slave follow

To instruct the subordinate group to inherit its state from a specified group, use the **hsrp slave follow** command in HSRP slave submode.

**follow** *mgo-session-name*

## Syntax Description

| | |
|---|---|
| *mgo-session-name* | Name of the MGO session from which the subordinate group will inherit the state. |

## Command Default

None

## Command Modes

HSRP Slave Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read, write |

### Example

This example shows how to instruct the subordinate group to inherit its state from a specified group.

```
Router# configure
Router(config)# router hsrp
Router(config-hsrp)# interface HundredGigE 0/4/0/4
Router(config-hsrp-if)# address-family ipv4
Router(config-hsrp-ipv4)# hsrp slave
Router(config-hsrp-slave)# follow m1
```

## Related Commands

| Command | Description |
|---|---|
| subordinate virtual mac address, on page 294 | Configures the virtual MAC address for the subordinate group. |

# subordinate primary virtual IPv4 address

To configure the primary virtual IPv4 address for the subordinate group, use the **subordinate primary virtual IPv4 address** command in the HSRP slave submode.

**address** *ip-address*

## Syntax Description

| | |
|---|---|
| *ip-address* | IP address of the Hot Standby router interface. |

## Command Default

None

## Command Modes

HSRP Slave Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read, write |

### Example

This example shows how to configure the primary virtual IPv4 address for the subordinate group.

```
Router# configure
Router(config)# router hsrp
Router(config-hsrp)# interface tenGigE 0/4/0/4
Router(config-hsrp-if)# address-family ipv4
Router(config-hsrp-ipv4)# hsrp slave
Router(config-hsrp-slave)# address 10.2.1.4
```

## Related Commands

| Command | Description |
|---|---|
| hsrp slave follow, on page 291 | Instructs the subordinate group to inherit its state from a specified group. |
| subordinate virtual mac address, on page 294 | Configures the virtual MAC address for the subordinate group. |

# subordinate secondary virtual IPv4 address

To configure the secondary virtual IPv4 address for the subordinate group, use the **subordinate secondary virtual IPv4 address** command in the HSRP slave submode.

**address** *ip-address* **secondary**

## Syntax Description

| | |
|---|---|
| *ip-address* | IP address of the Hot Standby router interface. |
| **secondary** | Sets the secondary hot standby IP address. |

## Command Default

None

## Command Modes

HSRP Slave Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read, write |

### Example

This example shows how to configure the secondary virtual IPv4 address for the subordinate group.

```
Router# configure
Router(config)# router hsrp
Router(config-hsrp)# interface HundredGigE 0/4/0/4
Router(config-hsrp-if)# address-family ipv4
Router(config-hsrp-ipv4)# hsrp slave
Router(config-hsrp-slave)# address 10.2.1.4 secondary
```

## Related Commands

| Command | Description |
|---|---|
| hsrp slave follow, on page 291 | Instructs the subordinate group to inherit its state from a specified group. |
| subordinate virtual mac address, on page 294 | Configures the virtual MAC address for the subordinate group. |

# subordinate virtual mac address

To configure the virtual MAC address for the subordinate group, use the **subordinate virtual mac address** command in the HSRP slave submode.

**mac-address** *address*

## Syntax Description

| | |
|---|---|
| *address* | 48-bit hardware address of ARP entry. |

## Command Default

None

## Command Modes

HSRP Slave Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| hsrp | read, write |

### Example

This example shows how to configure the virtual MAC address for the subordinate group.

```
Router# configure
Router(config)# router hsrp
Router(config-hsrp)# interface tenGigE 0/4/0/4
Router(config-hsrp-if)# address-family ipv4
Router(config-hsrp-ipv4)# hsrp slave
Router(config-hsrp-slave)# mac-address 10.2.4
```

## Related Commands

| Command | Description |
|---|---|
| hsrp slave follow, on page 291 | Instructs the subordinate group to inherit its state from a specified group. |

# timers (hsrp)

To configure the time between hello packets and the time before other routers declare the active Hot Standby or standby router to be down, use the **hsrp timers** command in HSRP group submode. To restore the timers to their default values, use the **no** form of this command.

**timers** {*hello-seconds* | **msec** *hello-milliseconds*} {*hold-seconds* | **msec** *hold-milliseconds*}
**no timers**

## Syntax Description

| | |
|---|---|
| hello-seconds | Hello interval in seconds. Range is from 1 to 255. Default is 3. |
| **msec** *hello-milliseconds* | Hello interval in milliseconds. Range is from 100 to 3000. |
| hold-seconds | Time in seconds before the active or standby router is declared to be down. Range is from 1 to 255. Default is 10. |
| **msec** *hold-milliseconds* | Time in milliseconds before the active or standby router is declared to be down. Range is from 100 to 3000. |

## Command Default

The default hello-seconds is 3. (If the **msec** keyword is specified, there is no default value.)

The default hold-seconds is 10. (If the **msec** keyword is specified, there is no default value.)

## Command Modes

HSRP Group Submode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Nonactive routers learn timer values from the active router, unless millisecond timer values are being used. If millisecond timer values are being used, all routers must be configured with the millisecond timer values. This rule applies if either the hello time or the hold time is specified in milliseconds.

The timers configured on the active router always override any other timer settings. All routers in a Hot Standby group should use the same timer values. Normally, the hold time is greater than or equal to three times the hello time (holdtime > 3 * hellotime).

You must specify either the *hello-seconds* argument or the **msec** keyword and *hello-milliseconds* argument, depending on whether you want the hello time in seconds or milliseconds. You must also specify either the *hold-seconds* argument or **msec** keyword and *hold-milliseconds* argument, depending on whether you want the hold time in seconds or milliseconds.

## Task ID

| Task ID | Operations |
|---|---|
| hsrp | read, write |

## Examples

This example shows how to set, for group number 1 on HundredGigE interface 0/2/0/1, the time between hello packets to 5 seconds and the time after which a router is considered to be down to 15 seconds. The configured timer values are used only if the router is active (or before they have been learned).

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 1
RP/0/RP0/CPU0:router(config-hsrp-gp)# timers 5 15
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

This example shows how to set, for group number 1 on HundredGigEinterface 0/2/0/1, the time between hello packets to 200 milliseconds and the time after which a router is considered to be down to 1000 milliseconds. The configured timer values are always used because milliseconds have been specified.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router hsrp
RP/0/RP0/CPU0:router(config-hsrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-hsrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-hsrp-ipv4)# hsrp 1 version 2
RP/0/RP0/CPU0:router(config-hsrp-gp)# timers msec 200 msec 1000
RP/0/RP0/CPU0:router(config-hsrp-gp)#
```

**Note**

- The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 2 for IPv6 address families.
- HSRP version 2 provides an extended group range of 0-4095.

## Related Commands

| Command | Description |
|---|---|
| show hsrp, on page 281 | Displays HSRP information. |

# LPTS Commands

This chapter describes the Cisco IOS XR software commands used to monitor Local Packet Transport Services (LPTS ).

For detailed information about LPTS concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# clear lpts ifib statistics

To clear the Internal Forwarding Information Base (IFIB) statistics, use the **clear lpts ifib statistics** command in XR EXEC mode.

**clear lpts ifib statistics location** *node-id*

## Syntax Description

| | |
|---|---|
| **location** *node-id* | Clears the IFIB statistics for the designated node. The *node-id* argument is entered in standard *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Task ID

| Task ID | Operations |
|---|---|
| lpts | execute |

## Examples

The following example shows how to clear the IFIB statistics for the RP:

```
RP/0/RP0/CPU0:router# clear lpts ifib statistics locatiion 0/0/cpu0
```

## Related Commands

| Command | Description |
|---|---|
| show lpts ifib statistics, on page 326 | Displays the LPTS IFIB statistics. |

# clear lpts pifib hardware statistics

To clear the Pre-Internal Forwarding Information Base (Pre-IFIB) hardware statistics, use the **clear lpts pifib hardware statistics** command in XR EXEC mode.

**clear lpts pifib hardware statistics** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Clears the Pre-IFIB hardware statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

If you do not specify a node with the **location** keyword and *node-id* argument, this command clears the Pre-IFIB hardware statistics for the node on which the command is run.

**Task ID**

| Task ID | Operations |
|---|---|
| lpts | execute |

**Examples**

The following example shows how to clear the Pre-IFIB hardware statistics for the RP:

```
RP/0/RP0/CPU0:router# clear lpts pifib hardware statistics location 0/0/CPU0
```

**Related Commands**

| Command | Description |
|---|---|
| show lpts pifib hardware policer, on page 342 | Displays the policer configuration value set. |

# clear lpts pifib statistics

To clear the Pre-Internal Forwarding Information Base (Pre-IFIB) statistics, use the **clear lpts pifib statistics** command in XR EXEC mode.

**clear lpts pifib statistics location** *node-id*

## Syntax Description

| | |
|---|---|
| **location** *node-id* | Clears the Pre-IFIB statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Task ID

| Task ID | Operations |
|---|---|
| lpts | execute |

## Examples

The following example shows how to clear the Pre-IFIB statistics for the RP:

```
RP/0/RP0/CPU0:router# clear lpts pifib statistics location 0/0/cpu0
```

## Related Commands

| Command | Description |
|---|---|
| show lpts pifib statistics, on page 348 | Displays the LPTS PIFIB statistics. |

# flow (LPTS)

To configure the policer for the Local Packet Transport Services (LPTS) flow type, use the **flow** command in pifib policer global configuration mode or pifib policer per-node configuration mode. To disable this feature, use the **no** form of this command.

**flow** *flow-type* **rate** *rate*
**no flow** *flow-type* **rate** *rate*

## Syntax Description

| | |
|---|---|
| flow-type | List of supported flow types. |
| **rate** *rate* | Specifies the rate in packets per seconds (PPS). The range is from 0 to 4294967295. |

## Command Default

The default behavior is to load the policer values from the static configuration file that is platform dependant.

## Command Modes

Pifib policer global configuration

Pifib policer per-node configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The table lists the supported flow types and the parameters that are used to define a policer.

***Table 38: List of Supported Flow Types***

| Flow Type | Description | Default Packet Rate (Recommended) |
|---|---|---|
| all-routers | Packets sent to all-routers multicast addresses, which include multicast LDP UDP packet. | 10000 |
| bgp-cfg-peer | Packets from a configured BGP peer. | 10000 |
| bgp-default | Packets from unconfigured, newly configured, or wildcard BGP peers. | 10000 |
| bgp-known | Packets from established BGP peering sessions. | 25000 |
| css-default | Packets from a new or newly established CSS session. | 1000 |
| css-known | Packets from an established CSS session. | 1000 |
| default-flow | Default flow type. | 500 |
| eigrp | EIGRP packets for configured interfaces. | 20000 |
| fragment | Fragmented packets. | 1000 |
| http-default | Packets from a new or newly established HTTP session. | 1000 |

| Flow Type | Description | Default Packet Rate (Recommended) |
|---|---|---|
| http-known | Packets from an established HTTP session. | 1000 |
| icmp-app | ICMP or ICMPv6 packets of interest to applications. | 2500 |
| icmp-control | ICPMv6 control packets. | 2500 |
| icmp-default | Other ICMP or ICMPv6 packets. | 2500 |
| icmp-local | ICMP or ICMPv6 packets with local interest. | 2500 |
| igmp | IGMP packets. | 3500 |
| ike | IKE packets. | 1000 |
| ipsec-default | AH or ESP packets with unknown or newly configured SPIs. | 1000 |
| ipsec-known | AH or ESP packets with known SPIs. | 3000 |
| isis-default | IS-IS packets for unconfigured (or newly, configured) interfaces. | 5000 |
| isis-known | IS-IS packets for configured interfaces. | 20000 |
| ldp-tcp-cfg-peer | Packets from a configured LDP TCP peer (SYNs or newly, established sessions). | 10000 |
| ldp-tcp-default | Packets from an unconfigured, newly configured, or wildcard LDP TCP peer. | 10000 |
| ldp-tcp-known | Packets from an established LDP peering session. | 25000 |
| ldp-udp | Unicast LDP UPD packets. | 500 |
| lmp-tcp-cfg-peer | Packets from a configured LMP TCP peer (SYNs or newly established sessions). | 10000 |
| lmp-tcp-default | Packets from an unconfigured, newly configured, or wild-card LMP TCP peer. | 10000 |
| lmp-tcp-known | Packets from an established LMP peering session. | 25000 |
| lmp-udp | Unicast LMP UDP packets. | 500 |
| msdp-cfg-peer | Packets from a configured MSDP peer. | 1000 |
| msdp-default | Packets from an unconfigured, newly configured, or wildcard MSDP peer. | 1000 |
| msdp-known | Packets from an established MSDP session. | 1000 |
| multicast-default | Packets for unconfigured or newly configured multicast groups. | 500 |
| multicast-known | Packets for configured multicast groups. | 25000 |

| Flow Type | Description | Default Packet Rate (Recommended) |
|---|---|---|
| ntp-known | Packets from an established NTP session. | 500 |
| ntp-default | Packets from a new or newly established NTP session. | 500 |
| ospf-mc_default | OSPF multicast packets for unconfigured (or newly configured) interfaces. | 5000 |
| ospf-mc-known | OSPF multicast packets for configured interfaces. | 20000 |
| ospf-uc-default | OSPF unicast packets for unconfigured (or newly configured) interfaces. | 1000 |
| ospf-uc-known | OSPF unicast packets for configured interfaces. | 5000 |
| pim-multicast | PIM multicast packets. | 23000 |
| pim-unicast | PIM unicast packets. | 10000 |
| rip | RIP packets. | 20000 |
| rsh-default | Packets from a new or newly established RSH session. | 1000 |
| rsh-known | Packets from an established RSH session. | 1000 |
| rsvp | RSVP packets. | 7000 |
| rsvp-udp | RSVP UDP packets. | 7000 |
| raw-default | Packets for unconfigured or newly configured IPv4 or IPv6 protocols. | 500 |
| raw-listen | Packets for configured IP protocols. | 500 |
| shttp-default | Packets from a new or newly established SSHTP session. | 1000 |
| shttp-known | Packets from an established SHTTP session. | 1000 |
| snmp | SNMP packets. | 2000 |
| ssh-default | Packets from a new or newly established SSH session. | 1000 |
| ssh-known | Packets from an established SSH session. | 1000 |
| tcp-cfg-peer | Packets for configured TCP peers. | 25000 |
| tcp-default | Packets for unconfigured or newly configured TCP services. | 500 |
| tcp-known | Packets for established TCP sessions. | 25000 |
| tcp-listen | Packets for configured TCP services. | 25000 |
| telnet-default | Packets from a new or newly established Telnet session. | 1000 |

| Flow Type | Description | Default Packet Rate (Recommended) |
|---|---|---|
| telnet-known | Packets from an established Telnet session. | 1000 |
| udp-cfg-peer | Packets for configured UDP-based protocol sessions. | 4000 |
| udp-default | Packets for unconfigured or newly configured UDP services. | 500 |
| udp-known | Packets for established UDP sessions. | 25000 |
| udp-listen | Packets for configured UDP services. | 4000 |

## Task ID

| Task ID | Operations |
|---|---|
| config-services | read, write |

## Examples

The following example shows how to configure the LPTS policer for the bgp-known flow type for all line cards:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# lpts pifib hardware police
RP/0/RP0/CPU0:router(config-pifib-policer-global)# flow bgp known rate 20000
```

The following example shows how to configure LPTS policer for the Intermediate System-to-Intermediate System (IS-IS)-known flow type for a specific line card:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:routerconfig)# lpts pifib hardware police location 0/2/CPU0
RP/0/RP0/CPU0:router(config-pifib-policer-per-node)# flow isis known rate 22222
```

# lpts pifib hardware police

To configure the ingress policers and to enter pifib policer global configuration mode or pifib policer per-node configuration mode, use the **lpts pifib hardware police** command in XR Config mode. To set the policer to the default value, use the **no** form of this command.

To map the LPTS policer with an ACL, use the **lpts pifib hardware police acl**

command in XR Config mode.

**lpts pifib hardware police** [ **location** *node-id*] [**flow** *flow-type* **rate** *rate*]
**no lpts pifib hardware police** [**location** *node-id* ] [**flow** *flow-type* **rate** *rate*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **flow** *flow-type* **rate** *rate* | LPTS flow type and the policer rate in packets per second (PPS). |

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read, write |
| config-services | read, write |

## Examples

This example shows how to configure the **lpts pifib hardware police** command for all line cards:

```
RP/0/RP0/CPU0:router(config)# lpts pifib hardware police
RP/0/RP0/CPU0:router(config-pifib-policer-global)#
```

This example shows how to configure the **lpts pifib hardware police** command for a specific line card:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# lpts pifib hardware police location 0/2/CPU0 flow dns rate
10
```

## Related Commands

| Command | Description |
|---|---|
| flow (LPTS), on page 301 | Configures the policer for the LPTS flow type. |
| show lpts pifib hardware policer, on page 342 | Displays the policer configuration value set. |

# lpts punt excessive-flow-trap

To enable the Excessive ARP Punt Protection feature and enter the control plane policer configuration mode, use the **lpts punt excessive-flow-trap** command in XR Config mode. To exit the control plane policer configuration mode and disable the Excessive ARP Punt Protection feature, use the **no** form of this command.

**lpts punt excessive-flow-trap** {**non-subscriber-interfaces** | **penalty-timeout**}
**no lpts punt excessive-flow-trap** {**non-subscriber-interfaces** | **penalty-timeout**}

## Syntax Description

| | |
|---|---|
| **non-subscriber-interfaces** | Enables the Excessive ARP Punt Protection for non-subscriber interfaces. |
| **penalty-timeout** | Sets the penalty timeout for a protocol. |

## Command Default

None

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.2.1 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| config-services | read, write |

## Examples

This example shows how to enable the Excessive ARP Punt Protection feature in the XR Config mode:

```
RP/0/RP0/CPU0:router(config)# lpts punt excessive-flow-trap
RP/0/RP0/CPU0:router(config-control-plane-policer)#
```

## Related Commands

| Command | Description |
|---|---|
| show running-config lpts punt excessive-flow-trap, on page 357 | Displays the running configuration for the Excessive Punt Flow Trap feature. |

# lpts punt excessive-flow-trap interface-based-flow

To enable interface-based flow (that is, considering all the packets received on a non-subscriber interface, irrespective of the source MAC address, to be a part of a single flow), use the **lpts punt excessive-flow-trap interface-based-flow** command in XR Config mode. To remove this interface-based flow configuration, use the **no** form of this command.

**lpts punt excessive-flow-trap interface-based-flow**
**no lpts punt excessive-flow-trap interface-based-flow**

## Syntax Description

This command has no keywords or arguments.

## Command Default

None

## Command Modes

## Command History

| Release | Modification |
|---|---|
| Release 5.2.5 | This command was introduced. |

## Usage Guidelines

Users cannot enable this command if Excessive Punt Flow Trap (EPFT) is turned on for the subscriber-interfaces and non-subscriber-interfaces MAC or vice versa. This is because, interface-based flow feature is mutually exclusive with MAC-based EPFT on non-subscriber interface feature.

## Task ID

| Task ID | Operation |
|---|---|
| config-services | read, write |

This example show how to enable interface-based flow:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#lpts punt excessive-flow-trap interface-based-flow
```

# lpts punt excessive-flow-trap non-subscriber-interfaces

To enable the Excessive ARP Punt Protection feature on non-subscriber interfaces, use the **lpts punt excessive-flow-trap non-subscriber-interfaces** command in XR Config mode. To disable the Excessive ARP Punt Protection feature on non-subscriber interfaces, use the **no** form of this command.

**lpts punt excessive-flow-trap non-subscriber-interfaces**
**no lpts punt excessive-flow-trap non-subscriber-interfaces**

## Syntax Description

This command has no keywords or arguments.

## Command Default

None

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 4.3.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| config-services | read, write |

## Examples

This example shows how to enable the Excessive ARP Punt Protection feature on the non-subscriber interfaces in the XR Config mode:

```
RP/0/RP0/CPU0:router(config)# lpts punt excessive-flow-trap non-subscriber-interfaces
RP/0/RP0/CPU0:router(config)#
```

## Related Commands

| Command | Description |
|---|---|
| show running-config lpts punt excessive-flow-trap, on page 357 | Displays the running configuration for the Excessive Punt Flow Trap feature. |

# lpts punt excessive-flow-trap penalty-timeout

To set the penalty timeout value for a protocol, use the **lpts punt excessive-flow-trap penalty-timeout** command in XR Config mode. To restore the default penalty timeout value, use the **no** form of this command.

**lpts punt excessive-flow-trap penalty-timeout arp** *timeout*
**no lpts punt excessive-flow-trap penalty-timeout arp** *timeout*

## Syntax Description

| | |
|---|---|
| *timeout* | Specify the penalty timeout value. |

## Command Default

The default penalty timeout value is 15 minutes.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.2.1 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| config-services | read, write |

## Examples

This example shows how to set the penalty timeout value of 70 minutes for the ARP protocol in the XR Config mode:

```
RP/0/RP0/CPU0:router(config)# lpts punt excessive-flow-trap penalty-timeout arp 70
RP/0/RP0/CPU0:router(config)#
```

## Related Commands

| Command | Description |
|---|---|
| show running-config lpts punt excessive-flow-trap, on page 357 | Displays the running configuration for the Excessive Punt Flow Trap feature. |

# show lpts bindings

To display the binding information in the Port Arbitrator, use the **show lpts bindings** command in XR EXEC mode.

**show lpts bindings** [**location** *node-id*] [**client-id** {**clnl** | **ipsec** | **ipv4-io** | **ipv6-io** | **mpa** | **tcp** | **test** | **udp** | **raw**}] [**brief**]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays information for the specified node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| client-id | (Optional) Type of client. It can be one of the following values:<br>• **clnl** —ISO connectionless protocol (used by IS-IS)<br>• **ipsec** —Secure IP<br>• **ipv4-io** —Traffic processed by the IPv4 stack<br>• **ipv6-io** —Traffic processed by the IPv6 stack<br>• **mpa** —Multicast Port Arbitrator (multicast group joins)<br>• **tcp** —Transmission Control Protocol<br>• **test** —Test applications<br>• **udp** —User Datagram Protocol<br>• **raw** —Raw IP |
| brief | (Optional) Displays summary output. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **show lpts bindings** command displays the Local Packet Transport Services (LPTS) bindings (requests to receive traffic of a particular type). Bindings are aggregated into flows by the LPTS Port Arbitrator; flows are then programmed into the Internal Forwarding Information Base (IFIB) and Pre-IFIB to direct packets to applications.

If you specify the optional **client-id** keyword and type of client, only bindings from that client are shown. If you specify the optional **location** keyword and *node-id* argument, only bindings from clients on that node are displayed.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts bindings** command, displaying bindings for all client ID types:

```
RP/0/RP0/CPU0:router# show lpts bindings

@ - Indirect binding; Sc - Scope

------------------------------------------
Location   :0/1/CPU0
Client ID  :IPV4_IO
Cookie     :0x00000001
Clnt Flags :
Layer 3    :IPV4
Layer 4    :ICMP
Local Addr :any
Remote Addr:any
Local Port :any
Remote Port:any
Filters    :Type / Intf or Pkt Type / Source Addr / Location
 INCLUDE_TYPE / type 8
 INCLUDE_TYPE / type 13
 INCLUDE_TYPE / type 17
------------------------------------------
Location   :0/2/CPU0
Client ID  :IPV4_IO
Cookie     :0x00000001
Clnt Flags :
Layer 3    :IPV4
Layer 4    :ICMP
Local Addr :any
Remote Addr:any
Local Port :any
Remote Port:any
Filters    :Type / Intf or Pkt Type / Source Addr / Location
 INCLUDE_TYPE / type 8
 INCLUDE_TYPE / type 13
 INCLUDE_TYPE / type 17
------------------------------------------
Location   :0/RP1/CPU0
Client ID  :TCP
Cookie     :0x4826f1f8
Clnt Flags :REUSEPORT
Layer 3    :IPV4
Layer 4    :TCP
Local Addr :any
Remote Addr:any
Local Port :7
Remote Port:any
------------------------------------------
Location   :0/RP1/CPU0
Client ID  :TCP
Cookie     :0x4826fa0c
Clnt Flags :REUSEPORT
Layer 3    :IPV4
Layer 4    :TCP
Local Addr :any
Remote Addr:any
Local Port :9
Remote Port:any
------------------------------------------
Location   :0/RP1/CPU0
Client ID  :TCP
Cookie     :0x482700d0
Clnt Flags :REUSEPORT
Layer 3    :IPV4
Layer 4    :TCP
```

```
Local Addr :any
Remote Addr:any
Local Port :19
Remote Port:any
----------------------------------------
Location   :0/RP1/CPU0
Client ID  :IPV4_IO
Cookie     :0x00000001
Clnt Flags :
Layer 3    :IPV4
Layer 4    :ICMP
Local Addr :any
Remote Addr:any
Local Port :any
Remote Port:any
Filters    :Type / Intf or Pkt Type / Source Addr / Location
 INCLUDE_TYPE / type 8
 INCLUDE_TYPE / type 13
 INCLUDE_TYPE / type 17
```

This table describes the significant fields shown in the display.

***Table 39: show lpts bindings Command Field Descriptions***

| Field | Description |
|---|---|
| Location | Node location, in the format of *rack/slot/module*. |
| Client ID | LPTS client type. |
| Cookie | Client's unique tag for the binding. |
| Clnt Flags | REUSEPORT -- client has set the SO_REUSEPORT or SO_REUSEADDR socket option. |
| Layer 3 | Layer 3 protocol (IPv4, IPv6, CLNL). |
| Layer 4 | Layer 4 protocol (TCP, UDP). |
| Local Addr | Local (destination) address. |
| Remote Addr | Remote (source) address. |
| Local Port | Local (destination) TCP or UDP port, or ICMP/IGMP packet type, or IPsec SPI. |
| Remote Port | Remote (source) TCP or UDP port. |

The following sample output is from the **show lpts bindings brief** command:

```
RP/0/RP0/CPU0:router# show lpts bindings brief

@ - Indirect binding; Sc - Scope

 Location   Clnt Sc L3   L4       Local,Remote Address.Port  Interface
 ---------- ---- -- ---- -----  -------------------------- ------------
 0/1/CPU0   IPV4 LO IPV4 ICMP          any.ECHO any               any
 0/1/CPU0   IPV4 LO IPV4 ICMP          any.TSTAMP any             any
 0/1/CPU0   IPV4 LO IPV4 ICMP          any.MASKREQ any            any
 0/1/CPU0   IPV6 LO IPV6 ICMP6         any.ECHOREQ any            any
```

```
0/3/CPU0    IPV4 LO IPV4 ICMP          any.ECHO any               any
0/3/CPU0    IPV4 LO IPV4 ICMP          any.TSTAMP any             any
```

This table describes the significant fields shown in the display.

***Table 40: show lpts bindings brief Command Field Descriptions***

| Field | Description |
|---|---|
| Location | Node location, in the format of *rack/slot/module*. |
| Clnt ID | LPTS client type. |
| Sc | Scope (LR = Logical-Router, LO = Local). |
| Layer 3 | Layer 3 protocol. |
| Layer 4 | Layer 4 protocol. |
| Local,Remote Address.Port | Local (destination) and Remote (source) addresses and ports or packet types. |
| Interface | Inbound interface. |

## Related Commands

| Command | Description |
|---|---|
| show lpts clients, on page 315 | Displays the client information for the Port Arbitrator. |
| show lpts flows, on page 317 | Displays information about LPTS flows. |

# show lpts clients

To display the client information for the Port Arbitrator, use the **show lpts clients** command in XR EXEC mode.

**show lpts clients** [**times**]

**Syntax Description**

| | |
|---|---|
| times | (Optional) Displays information about binding request rates and service times. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

The **show lpts clients** command displays the clients connected to the local packet transport services (LPTS) port arbitrator (PA).

**Task ID**

| Task ID | Operations |
|---|---|
| lpts | read |

**Examples**

The following sample output is from the **show lpts clients** command:

```
RP/0/RP0/CPU0:router# show lpts clients

o_flgs - open flags ; clid - client id
clid            loc              flags   o_flgs
RAW(3)          0/RP1/CPU0       0x1     0x2
TCP(1)          0/RP1/CPU0       0x1     0x2
IPV4_IO(5)      0/1/CPU0         0x3     0x2
IPV4_IO(5)      0/2/CPU0         0x3     0x2
IPV4_IO(5)      0/RP1/CPU0       0x3     0x2
MPA(7)          0/RP1/CPU0       0x3     0x0
```

This table describes the significant fields shown in the display.

***Table 41: show lpts clients Command Field Descriptions***

| Field | Description |
|---|---|
| Clid | LPTS client ID. |
| Loc | Node location, in the format *rack/slot/module*. |

| Field | Description |
|---|---|
| Flags | Client flags.<br>**Note** The client flags are used only for debugging purposes. |
| o_flags | Open flags.<br>**Note** The open flags are used only for debugging purposes. |

The following sample output is from the **show lpts clients times** command. The output shows samples for the last 30 seconds, 1 minute, 5 minutes, 10 minutes, and a total (if nonzero). The number of transactions, number of updates, and the minimum/average/maximum time in milliseconds to process each transaction is shown.

```
RP/0/RP0/CPU0:router# show lpts clients times

o_flgs - open flags ; clid - client id
clid            loc     flags   o_flgs
RAW(3)          0/RP1/CPU0      0x1     0x2
   30s:2 tx 2 upd 2/2/3ms/tx
    1m:2 tx 2 upd 2/2/3ms/tx
    5m:2 tx 2 upd 2/2/3ms/tx
   10m:2 tx 2 upd 2/2/3ms/tx
 total:2 tx 2 upd 2/-/3ms/tx
TCP(1)          0/RP1/CPU0      0x1     0x2
 total:3 tx 3 upd 1/-/1ms/tx
IPV4_IO(5)      0/1/CPU0        0x3     0x2
 total:1 tx 1 upd 0/-/0ms/tx
IPV4_IO(5)      0/2/CPU0        0x3     0x2
 total:1 tx 1 upd 1/-/1ms/tx
IPV4_IO(5)      0/RP1/CPU0      0x3     0x2
 total:1 tx 1 upd 3/-/3ms/tx
MPA(7)          0/RP1/CPU0      0x3     0x0
```

## Related Commands

| Command | Description |
|---|---|
| show lpts bindings, on page 311 | Displays the binding information in the port arbitrator. |
| show lpts flows, on page 317 | Displays information about LPTS flows. |

# show lpts flows

To display information about Local Packet Transport Services (LPTS) flows, use the **show lpts flows** command in XR EXEC mode.

**show lpts flows** [**brief**]

## Syntax Description

| | |
|---|---|
| brief | (Optional) Displays summary output. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **show lpts flows** command is used to display LPTS flows, which are aggregations of identical binding requests from multiple clients and are used to program the LPTS Internal Forwarding Information Base (IFIB) and Pre-IFIB.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts flows** command:

```
RP/0/RP0/CPU0:router# show lpts flows

-----------------------------------------
L3-proto    : IPV4(2)
L4-proto    : ICMP(1)

Local-IP    : any
Remote-IP   : any
Pkt-Type    : 8
Remote-Port : any
Interface   : any (0x0)
Flow-type   : ICMP-local
Min-TTL     : 0
Slice       : RAWIP4_FM
Flags       : 0x20 (in Pre-IFIB)
Location    : (drop)
Element References
location / count / scope
* / 3 / LOCAL
```

This table describes the significant fields shown in the display.

***Table 42: show lpts flows Command Field Descriptions***

| Field | Description |
|---|---|
| L3-proto | Layer 3 protocol (IPv4, IPv6, CLNL). |
| L4-proto | Layer 4 protocol (TCP, UDP, and so on). |
| Local-IP | Local (destination) IP address. |
| Remote-IP | Remote (source) IP address. |
| Pkt-Type | ICMP or IGMP packet type. |
| Remote-Port | Remote (source) TCP or UDP port. |
| Interface | Ingress interface. |
| Flow-type | Flow classification for hardware packet policing. |
| Min-TTL | Minimum time-to-live value expected from in the incoming packet. Any packet received with a lower TTL value will be dropped. |
| Slice | IFIB slice. |
| Flags | • Has FGID: Delivered to multiple destinations<br>• No IFIB entry: IFIB entry suppressed.<br>• Retrying FGID allocation<br>• In Pre-IFIB: Entry is in Pre-IFIB as well<br>• Deliver to one: If multiple bindings, will deliver to only one |
| Location | *rack/slot/module* to deliver to. |
| Element References | • location: *rack/slot/module* of client.<br>• count: number of clients at that location.<br>• scope: binding scope (LR:Logical Router, LOCAL:Local). |

The following sample output is from the **show lpts flows brief** command:

```
RP/0/RP0/CPU0:router# show lpts flows brief

+ - Additional delivery destination; L - Local interest; P - In Pre-IFIB

 L3   L4       Local, Remote Address.Port      Interface    Location    LP
 ---- -----  ------------------------------ ------------ ----------- --
 IPV4 ICMP          any.ECHO any                    any          (drop)      LP
 IPV4 ICMP          any.TSTAMP any                  any          (drop)      LP
 IPV4 ICMP          any.MASKREQ any                 any          (drop)      LP
 IPV6 ICMP6         any.ECHOREQ any                 any          (drop)      LP
 IPV4 any     224.0.0.2 any                  Gi0/1/0/1    0/5/CPU0     P
```

This table describes the significant fields shown in the display.

*Table 43: show lpts flows brief Command Field Descriptions*

| Field | Description |
|---|---|
| L3 | Layer 3 protocol (IPv4, IPv6, CLNL). |
| L4 | Layer 4 protocol. |
| Local, Remote Address.Port | Local (destination) and remote (source) IP addresses and TCP or UDP ports, or ICMP/IGMP packet types, or IPSec Security Parameters Indices. |
| Interface | Ingress interface. |
| Location | Delivery location:<br>• *rack/slot/module*—Individual location<br>• [0xNNNNN]—Multiple locations (platform-dependent value).<br>• (drop)—Do not deliver to any application |
| LP | Local interest (to be processed by IPv4 or IPv6 stack directly) or entry is resident in Pre-IFIB. |

## Related Commands

| Command | Description |
|---|---|
| show lpts bindings, on page 311 | Displays the binding information in the Port Arbitrator . |
| show lpts clients, on page 315 | Displays the client information for the Port Arbitrator . |

# show lpts ifib

To display the entries in the Internal Forwarding Information Base (IFIB), use the **show lpts ifib** command in XR EXEC mode.

**show lpts ifib** [**entry**] [{**type** {**bgp4** | **bgp6** | **isis** | **ospf-mc4** | **ospf-mc6** | **ospf4** | **ospf6** | **raw4** | **raw6** | **tcp4** | **tcp6** | **udp4** | **udp6**} | **all**}] [**brief** [**statistics**]] [**slices**] [**times**]

## Syntax Description

| | |
|---|---|
| entry | (Optional) Displays the IFIB entries. |
| **type** | (Optional) Displays the following protocol types.<br>• **bgp4** —IPv4 Border Gateway Protocol (BGP) slice<br>• **bgp6** —IPv6 BGP slice<br>• **isis** —Intermediate System-to-Intermediate System (IS-IS) slice<br>• **ospf-mc4** —IPv4 Open Shortest Path First (OSPF) multicast slice<br>• **ospf-mc6** —IPv6 OSPF multicast slice<br>• **ospf4** —IPv4 OSPF slice<br>• **ospf6** —IPv6 OSPF slice<br>• **raw4** —IPv4 raw IP<br>• **raw6** —IPv6 raw IP<br>• **tcp4** —IPv4 Transmission Control Protocol (TCP) slice<br>• **tcp6** —IPv6 TCP slice<br>• **udp4** —IPv4 UDP slice<br>• **udp6** —IPv6 UDP slice |
| all | Displays all IFIB types. |
| brief | (Optional) Displays the IFIB entries in brief format. |
| statistics | (Optional) Displays the IFIB table with statistics information. |
| slices | (Optional) Displays IFIB slices. |
| times | (Optional) Displays the IFIB update transaction times. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use this command to display detailed information about the entries in an IFIB slice. This command is useful for debugging problems with delivering packets to applications.

When the **statistics** keyword is used, detailed statistics are displayed for packet count, number of entries in each slice, and a total entries count.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts ifib** command:

```
RP/0/RP0/CPU0:router# show lpts ifib

O - Opcode; A - Accept Counter; D - Drop Counter; F - Flow Type; L - Listener Tag;
I - Local Flag; Y - SYN; T - Min TTL; DV - Deliver; DP - Drop; RE - Reassemble; na - Not
Applicable
------------------------------------------------------

Port/Type        : any
Source Port      : any
Dest IP          : any
Source IP        : any
Layer 4          : 88 (88)
Interface        : any (0x0)
O/A/D/F/L/I/Y/T  : DELIVER/0/0/EIGRP/IPv4_STACK/0/0/0
Deliver List     : 0/5/CPU0
------------------------------------------------------
```

This table describes the significant fields shown in the display.

***Table 44: show lpts ifib entries Command Field Descriptions***

| Field | Description |
|---|---|
| Port/Type | Destination (local) TCP or UDP port number, or ICMP/IGMP packet type, or IPSec Security Parameters Index.t2222 |
| Source Port | Source (remote) TCP or UDP port. |
| Dest IP | Destination (local) IP address. |
| Source IP | Source (remote) IP address. |
| Layer 4 | Layer 4 protocol number (6 = TCP).<br>**Note** Only the common Layer 4 protocol names are displayed. |
| Interface | Ingress interface name. |
| O/S/P/R/L/I/Y | • O: Opcode (DELIVER, DROP, or REASSEMBLE<br>• S: Stats counter<br>• P: Packet forwarding priority (LO, MED, or HIGH)<br>• R: Rate limit (LO, MED, or HIGH)<br>• L: Listener tag (IPv4_STACK, IPv6_STACK, or CLNL_STACK)<br>• I: Local-interest flag (0 or 1)<br>• Y: TCP SYN flag (0 or 1) |

| Field | Description |
|---|---|
| Deliver List | • (drop)—Drop packet<br>• *rack/slot/module*—Deliver to single destination<br>• [0xNNNN]—Deliver to multiple destinations (platform-dependent format) |

The following sample output is from the **show lpts ifib brief** command:

```
RP/0/RP0/CPU0:router# show lpts ifib brief

 Slice       L4     Interface    Dlvr        Local-Address,Port Remote-Address,Port
 --------  ------ ------------ ----------- ------------------------------------
 RAWIP4    UDP     any          0/0/CPU0    any 10.0.0.0/16
 RAWIP4    UDP     any          0/1/CPU0    any 10.32.0.0/16
 RAWIP4    IGMP    any          0/RP0/CPU0  any any
 RAWIP4     PIM    any          0/RP0/CPU0  any any
 RAWIP6     ICMP6  any          0/RP0/CPU0  any,MLDLQUERY any
 RAWIP6     ICMP6  any          0/RP0/CPU0  any,LSTNRREPORT any
 RAWIP6     ICMP6  any          0/RP0/CPU0  any,MLDLSTNRDN any
 RAWIP6     ICMP6  any          0/RP0/CPU0  any,LSTNRREPORT any
 RAWIP6     PIM    any          0/RP0/CPU0  any any
 RAWIP6     RAW    any          0/RP0/CPU0  any any
 UDP6       UDP    any          0/RP0/CPU0  any,547 any
```

The following sample output is from the **show lpts ifib brief statistics** command:

```
RP/0/RP0/CPU0:router# show lpts ifib brief statistics

 Slice       L4     Interface    Accept/Drop             Local-Address,Port
Remote-Address,Port
 -------- ------ ------------ ---------------------- ------------------------------------

 RAWIP4    UDP    any          0/0                    any 128.0.0.0/16
 RAWIP4    UDP    any          0/0                    any 128.32.0.0/16
 RAWIP4     IGMP   any          0/0                     any any
 RAWIP4     PIM    any          0/0                     any any
 RAWIP6     ICMP6  any          0/0                     any,MLDLQUERY any
 RAWIP6     ICMP6  any          0/0                     any,LSTNRREPORT any
 RAWIP6     ICMP6  any          0/0                     any,MLDLSTNRDN any
 RAWIP6     ICMP6  any          0/0                     any,LSTNRREPORT any
 RAWIP6     PIM    any          0/0                     any any
 RAWIP6     RAW    any          0/0                     any any
 UDP6       UDP    any          0/0                     any,547 any

 Slice    Num. Entries Accepts/Drops
 -------- ------------ -------------
 RAWIP4   4            0/0
 RAWIP6   6            0/0
 UDP6     1            0/0
 Total    11           0/0
```

## Related Commands

| Command | Description |
|---|---|
| show lpts ifib slices, on page 323 | Displays IFIB slice information. |

# show lpts ifib slices

To display Internal Forwarding Information Base (IFIB) slice information, use the **show lpts ifib slices** command in XR EXEC mode.

**show lpts ifib slices** [**type** {**bgp4** | **bgp6** | **isis** | **ospf-mc4** | **ospf-mc6** | **ospf4** | **ospf6** | **raw4** | **raw6** | **tcp4** | **tcp6** | **udp4** | **udp6**}] [**all**] [**statistics**] [**times**]

## Syntax Description

| | |
|---|---|
| type | (Optional) Enter protocol types.<br>• **bgp4** —IPv4 Border Gateway Protocol (BGP) slice<br>• **bgp6** —IPv6 BGP slice<br>• **isis** —Intermediate System-to-Intermediate System (IS-IS) slice<br>• **ospf-mc4** —IPv4 Open Shortest Path First (OSPF) multicast slice<br>• **ospf-mc6** —IPv6 OSPF multicast slice<br>• **ospf4** —IPv4 OSPF slice<br>• **ospf6** —IPv6 OSPF slice<br>• **raw4** —IPv4 raw IP<br>• **raw6** —IPv6 raw IP<br>• **tcp4** —IPv4 Transmission Control Protocol (TCP) slice<br>• **tcp6** —IPv6 TCP slice<br>• **udp4** —IPv4 UDP slice<br>• **udp6** —IPv6 UDP slice |
| all | (Optional) Displays all entries. |
| statistics | (Optional) Displays the statistics for slice lookups. |
| times | (Optional) Displays the IFIB update transaction times. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **show lpts ifib slices** command when troubleshooting IFIB entries and slice assignments. This command is especially useful when troubleshooting problems with delivering packets to applications.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts ifib slices** command:

```
RP/0/RP0/CPU0:router# show lpts ifib slices

Slice    L3   L4     Port  Location
-------- ---- ------ ----- --------
RAWIP4   IPV4 any    any   0/RP1/CPU0
RAWIP6   IPV6 any    any   0/RP1/CPU0
OSPF4    IPV4 OSPF   any   0/RP1/CPU0
OSPF6    IPV6 OSPF   any   0/RP1/CPU0
OSPF_MC4 IPV4 any    any   0/RP1/CPU0
OSPF_MC6 IPV6 any    any   0/RP1/CPU0
BGP4     IPV4 TCP    179   0/RP1/CPU0
BGP6     IPV6 TCP    179   0/RP1/CPU0
L2TP4    IPV4 UDP    1701  0/RP1/CPU0
UDP4     IPV4 UDP    any   0/RP1/CPU0
UDP6     IPV6 UDP    any   0/RP1/CPU0
TCP4     IPV4 TCP    any   0/RP1/CPU0
TCP6     IPV6 TCP    any   0/RP1/CPU0
ISIS     CLNS -      any   0/RP1/CPU0
```

The following sample output is from the **show lpts ifib slices times** command:

```
RP/0/RP0/CPU0:router# show lpts ifib slices times

Slice    L3   L4     Port  Location
-------- ---- ------ ----- --------
RAWIP4   IPV4 any    any   0/RP1/CPU0
RAWIP6   IPV6 any    any   0/RP1/CPU0
OSPF4    IPV4 OSPF   any   0/RP1/CPU0
OSPF6    IPV6 OSPF   any   0/RP1/CPU0
OSPF_MC4 IPV4 any    any   0/RP1/CPU0
OSPF_MC6 IPV6 any    any   0/RP1/CPU0
BGP4     IPV4 TCP    179   0/RP1/CPU0
BGP6     IPV6 TCP    179   0/RP1/CPU0
L2TP4    IPV4 UDP    1701  0/RP1/CPU0
UDP4     IPV4 UDP    any   0/RP1/CPU0
UDP6     IPV6 UDP    any   0/RP1/CPU0
TCP4     IPV4 TCP    any   0/RP1/CPU0
TCP6     IPV6 TCP    any   0/RP1/CPU0
ISIS     CLNS -      any   0/RP1/CPU0

  Flow Manager 0/RP1/CPU0:
   total:5 tx 13 upd 1/-/1ms/tx
```

The following sample output is from the **show lpts ifib slices statistics** command:

```
RP/0/RP0/CPU0:router# show lpts ifib slices all statistics

Slice    L3   L4     Port  Location   Lookups RmtDlvr Rejects RLDrops NoEntry
-------- ---- ------ ----- ---------- ------- ------- ------- ------- -------
RAWIP4   IPV4 any    any   0/0/CPU0   5       0       0       0       0
RAWIP6   IPV6 any    any   0/0/CPU0   0       0       0       0       0
OSPF4    IPV4 OSPF   any   0/0/CPU0   0       0       0       0       0
OSPF6    IPV6 OSPF   any   0/0/CPU0   0       0       0       0       0
OSPF_MC4 IPV4 any    any   0/0/CPU0   0       0       0       0       0
OSPF_MC6 IPV6 any    any   0/0/CPU0   0       0       0       0       0
BGP4     IPV4 TCP    179   0/0/CPU0   0       0       0       0       0
BGP6     IPV6 TCP    179   0/0/CPU0   0       0       0       0       0
L2TP4    IPV4 UDP    1701  0/0/CPU0 0       0       0       0       0
UDP4     IPV4 UDP    any   0/0/CPU0   3704    0       979     0       0
UDP6     IPV6 UDP    any   0/0/CPU0   0       0       0       0       0
```

```
TCP4     IPV4 TCP    any   0/0/CPU0   0       0       0       0       0
TCP6     IPV6 TCP    any   0/0/CPU0   0       0       0       0       0
ISIS     CLNS -      any   0/0/CPU0   0       0       0       0       0

  Flow Manager 0/0/CPU0:
   Packets in: 3792
   Packets delivered locally without lookups: 83
   Slice lookups: 3709
     Rejects: 979
```

This table describes the significant fields shown in the display.

***Table 45: show lpts ifib slices statistics Command Field Descriptions***

| Field | Description |
|---|---|
| Slice | Slice number. |
| L3-proto | Layer 3 protocol (IPv4, IPv6, CLNL). |
| L4-proto | Layer 4 protocol (TCP, UDP, and others). |
| Port | Local (destination) TCP or UDP port. |
| Location | Node location, in the format *rack/slot/module*. |

## Related Commands

| Command | Description |
|---|---|
| show lpts ifib , on page 320 | Displays entries in the IFIB. |

# show lpts ifib statistics

To display Internal Forwarding Information Base (IFIB) statistics, use the **show lpts ifib statistics** command in XR EXEC mode.

**show lpts ifib statistics** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays IFIB statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

No default behavior or values

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operations |
|---|---|
| lpts | read |

**Examples**

The following sample output is from the **show lpts ifib statistics** command:

```
RP/0/RP0/CPU0:router# show lpts ifib statistics

Flow Manager 0/RP1/CPU0:
   Packets in:254
   Packets delivered locally without lookups:0
   Slice lookups:254

    Packets delivered locally:0
    Packets delivered remotely:0
```

This table describes the significant fields shown in the display.

***Table 46: show lpts ifib statistics Command Field Descriptions***

| Field | Description |
|---|---|
| Packets in | Packets presented to the LPTS decaps node in netio. |
| Packets delivered locally without lookups | Packets previously resolved on a LC delivered directly to L3. |
| Slice lookups | Packets requiring slice lookups. |
| Post-lookup error drops | Packets dropped after a slice lookup. |

| Field | Description |
|---|---|
| Rejects | Packets that caused a TCP RST or ICMP Port/Protocol Unreachable. |
| Packets delivered locally | Packets delivered to local applications after slice lookups. |
| Packets delivered remotely | Packets delivered to applications on remote RPs. |

**Note** The sample output is an example only and displays only those fields showing a value. No display exists for nonzero values. This command may show other values depending on your router configuration.

## Related Commands

| Command | Description |
|---|---|
| show lpts ifib , on page 320 | Displays the entries in an IFIB slice. |

# show lpts ifib times

To display Internal Forwarding Information Base (IFIB) update transaction times, use the **show lpts ifib times** command in XR EXEC mode.

**show lpts ifib times** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays IFIB update transaction times for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts ifib times** command:

```
RP/0/RP0/CPU0:router# show lpts ifib times

Slice    L3   L4     Port  Location
-------- ---- ------ ----- --------
RAWIP4   IPV4 any    any   0/RP1/CPU0
RAWIP6   IPV6 any    any   0/RP1/CPU0
OSPF4    IPV4 OSPF   any   0/RP1/CPU0
OSPF6    IPV6 OSPF   any   0/RP1/CPU0
OSPF_MC4 IPV4 any    any   0/RP1/CPU0
OSPF_MC6 IPV6 any    any   0/RP1/CPU0
BGP4     IPV4 TCP    179   0/RP1/CPU0
BGP6     IPV6 TCP    179   0/RP1/CPU0
UDP4     IPV4 UDP    any   0/RP1/CPU0
UDP6     IPV6 UDP    any   0/RP1/CPU0
TCP4     IPV4 TCP    any   0/RP1/CPU0
TCP6     IPV6 TCP    any   0/RP1/CPU0
ISIS     CLNS -      any   0/RP1/CPU0
MCAST4   IPV4 any    any   0/RP1/CPU0
MCAST6   IPV6 any    any   0/RP1/CPU0
Flow Manager 0/RP1/CPU0:
 total:5 tx 13 upd 1/-/1ms/tx
```

This table describes the significant fields shown in the display.

***Table 47: show lpts ifib times Command Field Descriptions***

| Field | Description |
|---|---|
| Slice | Slice number. |

| Field | Description |
|---|---|
| L3 Protocol | Layer 3 protocol (IPv4, IPV6, CLNL). |
| L4 Protocol | Layer 4 protocol (TCP, UDP, and so on). |
| Port | Local (destination) TCP or UDP port. |
| Location | Node location, in the format *rack/slot/module*. |

## Related Commands

| Command | Description |
|---|---|
| show lpts ifib , on page 320 | Displays detailed information about entries in an IFIB slice. |

# show lpts mpa groups

To display aggregate information about multicast bindings for groups, use the **show lpts mpa groups** command in XR EXEC mode.

**show lpts mpa groups** *type interface-path-id*

## Syntax Description

| | |
|---|---|
| type | Interface type. For more information, use the question mark (?) online help function. |
| interface-path-id | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the modular services card or line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/ RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **show lpts mpa groups** command is used to aggregate information about the multicast groups joined on a specified interface. This command also displays the filter mode and source list associated with the groups joined on a specified interface.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |
| network | read |

## Examples

The following sample output is from the **show lpts mpa groups** command:

```
RP/0/RP0/CPU0:router# show lpts mpa groups HundredGigE0/0/0/0
```

```
224.0.0.2 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
224.0.0.13 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
224.0.0.22 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
```

This table describes the significant fields shown in the display.

***Table 48: show lpts mpa groups Command Field Descriptions***

| Field | Description |
|---|---|
| Includes | Displays the number of sockets that have set up an INCLUDE mode filter for that group and if there are any source-specific filters. |
| Excludes | Displays the number of sockets that have set up an EXCLUDE mode filter for that group and if there are any source-specific filters. |

# show lpts pifib

To display Pre-Internal Forwarding Information Base (Pre-IFIB) entries, use the **show lpts pifib** command in XR EXEC mode.

**show lpts pifib** [**entry**][**hardware** {**entry** | **police**}] [**type**{**isis** | **ipv4** | **ipv6** }] {**any** | **frag**} [**brief**] **[statistics][location** *node-id*]

## Syntax Description

| | |
|---|---|
| entry | (Optional) Pre-IFIB entry. |
| hardware | (Optional) Displays hardware for Pre-IFIB. |
| entry | (Optional) Displays the entries for Pre-IFIB. |
| police | (Optional) Displays the policer values that are being use. |
| type | (Optional) Protocol type. |
| isis | Intermediate System-to-Intermediate System (IS-IS) sub Pre-IFIB type. |
| ipv4 | IPv4 sub Pre-IFIB type. Possible values include **frag** , **ixmp** , **mcast** , **tcp** , **udp** , **ipsec** , and **raw** . |
| ipv6 | IPv6 sub Pre-IFIB type. Possible values include **frag** , **icmp** , **ixmp** , **mcast** , **tcp** , **udp** , **ipsec, and raw** . |
| any | Any IPv4 or IPv6 protocol. |
| brief | (Optional) Pre-IFIB entries in brief format. |
| statistics | (Optional) Pre-IFIB table with statistics information. |
| **location** *node-id* | (Optional) The *node-id* argument is entered in the *rack/slot/module* notation (for example, 0/7/CPU0). |

## Command Default

By default, all entries are displayed.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **show lpts pifib** command with the **brief** keyword to perform the following functions:

- Display entries of all or part of a Pre-IFIB.
- Display a short description of each entry in the LPTS Pre-IFIB, optionally displaying packet counts for each entry.

**Note** These statistics are used only for packets that are processed by a line card, route processor, or distributed route processor.

Pre-IFIB statistics for packets processed by line card hardware are counted separately.

By default, all the defaults are displayed.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following is sample output for the **show lpts pifib** command:

```
RP/0/RP0/CPU0:router# show lpts pifib

O - Opcode; F - Flow Type; L - Listener Tag; I - Local Flag; T - Min TTL;
na - Not Applicable
-----------------------------------------------------
L3 Protocol       : CLNS
L4 Protocol       : -
VRF-ID            : default (0x60000000)
Destination IP    : any
Source IP         : any
Port/Type         : any
Source Port       : any
Is Fragment       : 0
Is SYN            : 0
Interface         : any (0x0)
O/F/L/I/T         : DELIVER/ISIS-default/CLNS_STACK/0/0
Deliver List      : FGID 11935
Accepts/Drops     : 0/0
Is Stale          : 0
```

The following is sample output for the **show lpts pifib type** command using the **ipv4** and **frag** keywords.

```
RP/0/RP0/CPU0:router# show lpts pifib type ipv4 frag

O - Opcode; F - Flow Type; L - Listener Tag; I - Local Flag; T - Min TTL;
na - Not Applicable
-----------------------------------------------------
L3 Protocol       : IPV4
L4 Protocol       : any
Destination IP    : any
Source IP         : any
Port/Type         : any
Source Port       : any
Is Fragment       : 1
Is SYN            : 0
Interface         : any (0x0)
O/F/L/I/T         : REASSEMBLE/Fragment/IPv4_REASS/0/0
Deliver List      :
Accepts/Drops     : 0/0
Is Stale          : 0
```

The following is sample output from the **show lpts pifib** command with the **entry** and **brief** keywords added :

```
RP/0/RP0/CPU0:router# show lpts pifib entry brief
```

```
* - Critical Flow; I - Local Interest;
X - Drop; R - Reassemble;

 Type          Local, Remote Address.Port       L4    Interface    Deliver
 ----------  -------------------------------   ----- ------------ ----------

 ISIS               - -                         -     any          0/0/CPU0
 IPv4_frag          any any                     any   any          R
 IPv4_IXMP          any.ECHO any                ICMP  any          XI
 IPv4_IXMP          any.TSTAMP any              ICMP  any          XI
 IPv4_IXMP          any.MASKREQ any             ICMP  any          XI
 IPv4_IXMP          any any                     ICMP  any          0/0/CPU0
 IPv4_IXMP          any any                     IGMP  any          0/0/CPU0
 IPv4_mcast         224.0.0.5 any               any   any          0/0/CPU0
 IPv4_mcast         224.0.0.6 any               any   any          0/0/CPU0
 IPv4_mcast         224.0.0.0/4 any             any   any          0/0/CPU0

 IPv4_TCP           any.179 any                 TCP   any          0/0/CPU0
 IPv4_TCP           any any.179                 TCP   any          0/0/CPU0
 IPv4_TCP           any any                     TCP   any          0/0/CPU0
 IPv4_UDP           any any                     UDP   any          0/0/CPU0
 IPv4_IPsec         any any                     ESP   any          0/0/CPU0
 IPv4_IPsec         any any                     AH    any          0/0/CPU0
 IPv4_rawIP         any any                     OSPF  any          0/0/CPU0
 IPv4_rawIP         any any                     any   any          0/0/CPU0
 IPv6_frag          any any                     any   any          R
 IPv6_ICMP          any.na any                  ICMP6 any          XI
 IPv6_ICMP          any any                     ICMP6 any          0/0/CPU0
 IPv6_mcast         ff02::5 any                 any   any          0/0/CPU0
 IPv6_mcast         ff02::6 any                 any   any          0/0/CPU0
 IPv6_mcast         ff00::/8 any                any   any          0/0/CPU0
 IPv6_TCP           any.179 any                 TCP   any          0/0/CPU0
 IPv6_TCP           any any.179                 TCP   any          0/0/CPU0
 IPv6_TCP           any any                     TCP   any          0/0/CPU0
 IPv6_UDP           any any                     UDP   any          0/0/CPU0
 IPv6_IPsec         any any                     ESP   any          0/0/CPU0
 IPv6_IPsec         any any                     AH    any          0/0/CPU0
 IPv6_rawIP         any any                     OSPF  any          0/0/CPU0
 IPv6_rawIP         any any                     any   any          0/0/CPU0
```

The following sample output is from the **show lpts pifib** command with the **entry, brief,** and **statistics** keywords added :

```
RP/0/RP0/CPU0:router# show lpts pifib entry brief statistics

* - Critical Flow; I - Local Interest;
X - Drop; R - Reassemble;

 Type         Local, Remote Address.Port      L4    Interface    Accepts/Drops

 ----------  ---------------------------   ----- ------------ --------------

 ISIS               - -                         -     any          0/0
 IPv4_frag          any any                     any   any          0/0
 IPv4_IXMP          any.ECHO any                ICMP  any          0/0
 IPv4_IXMP          any.TSTAMP any              ICMP  any          0/0
 IPv4_IXMP          any.MASKREQ any             ICMP  any          0/0
 IPv4_IXMP          any any                     ICMP  any          5/0
 IPv4_IXMP          any any                     IGMP  any          0/0
 IPv4_mcast         224.0.0.5 any               any   any          0/0
 IPv4_mcast         224.0.0.6 any               any   any          0/0
```

```
 IPv4_mcast         224.0.0.0/4 any           any   any          0/0
 IPv4_TCP           any.179 any               TCP   any          0/0
 IPv4_TCP           any any.179               TCP   any          0/0
 IPv4_TCP           any any                   TCP   any          0/0
 IPv4_UDP           any any                   UDP   any          4152/0
 IPv4_IPsec         any any                   ESP   any          0/0
 IPv4_IPsec         any any                   AH    any          0/0
 IPv4_rawIP         any any                   OSPF  any          0/0


-----------------------


statistics:

Type            Num. Entries         Accepts/Drops

------          ------------         -------------
ISIS            1                    0/0
IPv4_frag       1                    0/0
IPv4_IXMP       5                    5/0
IPv4_mcast      3                    0/0
IPv4_TCP        3                    0/0
IPv4_UDP        1                    4175/0
IPv4_IPsec      2                    0/0
IPv4_rawIP      2                    0/0
IPv6_frag       1                    0/0
IPv6_ICMP       2                    0/0
IPv6_mcast      3                    0/0
IPv6_TCP        3                    0/0
IPv6_UDP        1                    0/0
IPv6_IPsec      2                    0/0
IPv6_rawIP      2                    0/0
Total           32

Packets into Pre-IFIB: 4263
Lookups: 4263
Packets delivered locally: 4263
Packets delivered remotely: 0
```

This table describes the significant fields shown in the display for the **show lpts pifib brief statistics** .

***Table 49: show lpts pifib Command Field Descriptions***

| Field | Description |
|---|---|
| Type | Hardware entry type. |
| Local, Remote Address. Port | Indicates local address (in the form of local port and type) and remote address (remote port). |
| L4 | Layer 4 protocol of the entry. |
| Interface | Interface for this entry. |
| Accepts/Drops | Number of packets sent to DestAddr/Number of packets dropped due to policing. |
| Num. Entries | Number of pre-ifib entries of the listed type. |

| Field | Description |
|---|---|
| Packets into Pre-IFIB | Packets presented for pre-IFIB lookups. |
| Lookups | Packets looked up. |
| Packets delivered locally | Packets delivered to local applications or the local stack (*n* duplicated) packets duplicated for delivery to applications and the local stack. |
| Packets delivered remotely | Packets delivered to applications or for lookup on other RPs. |

# show lpts pifib hardware context

To display the context for the Local Packet Transport Services (LPTS) pre-IFIB hardware-related data structures, use the **show lpts pifib hardware context** command in XR EXEC mode.

**show lpts pifib hardware context** [**location** {**all** | }]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays pre-Internal Forwarding Information Base (IFIB) information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| all | Specifies all locations. |

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts pifib hardware context** command with the **location** keyword:

```
RP/0/RP0/CPU0:router# show lpts pifib hardware context location 0/0/cpu0

             Node: 0/0/CPU0:
----------------------------------------------
Initialization phase: Done
Hash table levels: 1
Hash table buckets per level: 1021
Total number of Hardware policers: 100
Total number of inuse Hardware policers: 82
The start offset of Hardware policers: 109
----------------------------------------------
NPU 0:
----------------------------
 IPv4 Region:
 # of Hardware Insert: 0
 # of Hardware Delete: 0
 # of Hardware Update: 0
 # of Hardware Entry : 21
 # of  1-entry Bucket: 21
----------------------------
 IPv6 Region:
 # of Hardware Insert: 0
 # of Hardware Delete: 0
 # of Hardware Update: 0
 # of Hardware Entry : 26
 # of  1-entry Bucket: 26
```

```
----------------------------
 ISIS Region:
 # of Hardware Insert: 0
 # of Hardware Delete: 0
 # of Hardware Update: 0
 # of Hardware Entry : 1
 # of  1-entry Bucket: 1
----------------------------
 BFD Region:
 # of Hardware Insert: 0
 # of Hardware Delete: 0
 # of Hardware Update: 0
 # of Hardware Entry : 3
 # of  1-entry Bucket: 3
```

# show lpts pifib hardware entry

To display entries in the Local Packet Transport Services (LPTS) pre-IFIB hardware table, use the **show lpts pifib hardware entry** command in XR EXEC mode.

**show lpts pifib hardware entry** [**acl** *acl-name*] [**type** {**bfd** | **ipv4** | **ipv6** | **isis**}] [**start-index** *number* **num-entries** *number*] [{**brief** | **statistics**}] [**location** {**all***node_id*}]

## Syntax Description

| | |
|---|---|
| type | (Optional) Specifies the hardware entry type. Enter one of the following types:<br>• **ipv4** —Specifies IPv4 entries.<br>• **ipv6** —Specifies IPv6 entries.<br>• **isis** —Specifies ISIS entries.<br>• **bfd**—Specifies BFD entries. |
| **start-index** *number* | (Optional) Starting index number. |
| **num-entries** *number* | (Optional) Maximum entries permitted. |
| brief | (Optional) Displays summary hardware entry information. |
| statistics | (Optional) Displays hardware entry accept or drop statistics for each summary entry. |
| all | Specifies all locations. |

## Command Default

Displays hardware entry information in brief.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

SNMP is not supported on ASR 9000 4th Generation Line Cards, Therefore, the ACLs that are configured on ASR 9000 4th Generation Line Cards are not displayed by running this command.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts pifib hardware entry** command with the **location** keyword:

```
RP/0/RP0/CPU0:router# show lpts pifib hardware entry location 0/1/CPU0

Node: 0/0/CPU0:
----------------------------------------
M - Fabric Multicast;
L - Listener Tag; T - Min TTL;
```

```
F - Flow Type;
DestNode - Destination Node;
DestAddr - Destination Fabric queue;
SID - Stream ID;
Po - Policer; Ct - Stats Counter;
Lp - Lookup priority; Sp - Storage Priority;
Ar - Average rate limit; Bu - Burst;
HAr - Hardware Average rate limit; HBu - Hardware Burst;
Cir - Committed Information rate in HAL
Rsp - Relative sorting position;
Rtp - Relative TCAM position;
na - Not Applicable or Not Available
-----------------------------------------------------

Destination IP    : any
Source IP         : any
Is Fragment       : 0
Interface         : any
M/L/T/F           : 0/ISIS_FM/0/ISIS-default
DestNode          : 48
DestAddr          : 48
SID               : 9
L4 Protocol       : -
Source port       : any
Destination Port  : any
Ct                : 0xd84da
Accepted/Dropped  : 0/0
Lp/Sp             : 0/0
# of TCAM entries : 1
HPo/HAr/HBu/Cir   : 1879638/2000pps/2000ms/2000pps
State             : Entry in TCAM
Rsp/Rtp           : 0/2

             Node: 0/1/CPU0:
----------------------------------------
V - Vital; M - Fabric Multicast;
C - Moose Congestion Flag; L - Listener Tag; T - Min TTL;
F - Flow Type;
DestNode - Destination Node;
DestAddr - Destination Fabric Address;
Sq - Ingress Shaping Queue; Dq - Destination Queue;
Po - Policer; Ct - Stats Counter;
Lp - Lookup priority; Sp - Storage Priority;
Ar - Average rate limit; Bu - Burst;
Rsp - Relative sorting position;
-----------------------------------------------------
L4 Protocol       : any

Source IP         : any
Port/Type         : any
Source Port       : any
Is Fragment       : 1
Is SYN            : any
Interface         : any
V/M/C/L/T/F       : 0/0/0/IPv4_REASS/0/Fragment
DestNode          : Local
DestAddr          : Punt
Sq/Dq/Ct          : 4/na/0x24400
Accepted/Dropped  : 0/0
Lp/Sp             : 0/0
# of TCAM entries : 1
Po/Ar/Bu          : 101/1000pps/100ms
State             : Entry in TCAM
```

```
Rsp/Rtp           : 0/0
-----------------------------------------------------
```

This table describes the significant fields shown in the display.

***Table 50: show lpts pifib hardware entry Command Field Descriptions***

| Field | Description |
|---|---|
| L4 Protocol | Layer 4 protocol of the entry. |
| Source IP | Source IP address for this entry. |
| Port/Type | Port or type for this entry. |
| Source Port | Source port for this entry. |
| Is Fragment | Indicates if this entry applies to IP fragments. |
| Is SYN | Indicates if this entry applies to TCP SYNs. |
| Interface | Interface for this entry. |
| V/M/C/L/T/F | • V—vital<br>• M—fabric multicast<br>• C—moose congestion flag<br>• L—listener tag<br>• T—minimum time-to-live<br>• F—flow type |
| DestNode | Destination node to which to send the packet. |
| DestAddr | Destination address to which to send the packet. |
| Sq/Dq/Ct | • Sq—Ingress Shaping Queue<br>• Dq—Destination Queue<br>• Ct—Stats Counter |
| Accepted/Dropped | Number of packets sent to DestAddr/Number of packets dropped due to policing. |

# show lpts pifib hardware policer

Displays all the LPTS policer entries from the pre-Internal Forwarding Information Base (PIFIB).

**show lpts pifib hardware policer** [**location** {**all***node-id*}]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays pre-Internal Forwarding Information Base (IFIB) information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| all | Specifies all locations. |

## Command Default

If no policer is configured, the default value is the configured rate.

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

To retrieve command outputs, the **flow monitor-map** and **sampler-map** statements must be configured and applied to the respective interface, as shown in the following example:

```
!
flow monitor-map fmm
record ipv4
cache entries 10000
cache timeout active 15
cache timeout inactive 5
!
sampler-map fsm
random 1 out-of 1
!
interface MgmtEth0/RSP0/CPU0/0
ipv4 address 10.20.10.10 255.255.0.0
!
interface TenGigE0/3/0/0
ipv4 address 192.168.1.1 255.255.255.0
flow ipv4 monitor fmm sampler fsm ingress
flow ipv4 monitor fmm sampler fsm egress
ipv4 access-group SLMN-DPI ingress
!
```

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

This sample output is from the **show lpts pifib hardware policer** command with the **location** keyword for 0/2/CPU0:

The XML form of the output can be retrieved as follows:

```
RP/0/RP0/CPU0:router# show operational platformLPTSPIfib
NodeTable node/NodeName/Rack=0;Slot=2;Instance=CPU0 Police xml
```

```
...
<?xml version="1.0"?>
<Response MajorVersion="1" MinorVersion="0">
    <Get>
        <Operational>
            <PlatformLPTSPIfib MajorVersion="0" MinorVersion="0">
                <NodeTable>
                    <Node>
                        <Naming>
                            <NodeName>
                                <Rack>
                                    0
                                </Rack>
                                <Slot>
                                    2
                                </Slot>
                                <Instance>
                                    CPU0
                                </Instance>
                            </NodeName>
                        </Naming>
                        <Police>
                            <police_info>
                                <Entry>
                                    <avgrate>
                                        2500
                                    </avgrate>
                                    <burst>
                                        1250
                                    </burst>
                                    <static_avgrate>
                                        2500
                                    </static_avgrate>
                                    <avgrate_type>
                                        Static
                                    </avgrate_type>
                                    <flow_type>
                                        unconfigured-default
                                    </flow_type>
                                    <accepted_stats>
                                        0
                                    </accepted_stats>
                                    <dropped_stats>
                                        0
                                    </dropped_stats>
                                    <policer>
                                        0
                                    </policer>
                                    <iptos_value>
                                        0
                                    </iptos_value>
                                    <change_type>
                                        0
                                    </change_type>
                                    <acl_config>
                                        0
                                    </acl_config>
                                    <acl_str>

                                    </acl_str>
                                    <np>
                                        0
                                    </np>
                                </Entry>
```

```
<Entry>
    <avgrate>
        10000
    </avgrate>
    <burst>
        5000
    </burst>
    <static_avgrate>
        10000
    </static_avgrate>
    <avgrate_type>
        Static
    </avgrate_type>
    <flow_type>
        L2TPv2-fragment
    </flow_type>
    <accepted_stats>
        0
    </accepted_stats>
    <dropped_stats>
        0
    </dropped_stats>
    <policer>
        85
    </policer>
    <iptos_value>
        0
    </iptos_value>
    <change_type>
        0
    </change_type>
    <acl_config>
        0
    </acl_config>
    <acl_str>

    </acl_str>
    <np>
        0
    </np>
</Entry>
<Entry>
    <avgrate>
        2500
    </avgrate>
    <burst>
        1250
    </burst>
    <static_avgrate>
        2500
    </static_avgrate>
    <avgrate_type>
        Static
    </avgrate_type>
    <flow_type>
        Fragment
    </flow_type>
    <accepted_stats>
        0
    </accepted_stats>
    <dropped_stats>
        0
    </dropped_stats>
    <policer>
```

```
                                        1
                                    </policer>
                                    <iptos_value>
                                        0
                                    </iptos_value>
                                    <change_type>
                                        0
                                    </change_type>
                                    <acl_config>
                                        0
                                    </acl_config>
                                    <acl_str>

                                    </acl_str>
                                    <np>
                                        0
                                    </np>
                                </Entry>
...

RP/0/RP0/CPU0:router# show lpts pifib hardware policer location 0/2/CPU0

             Node: 0/2/CPU0:
----------------------------------------
flow_type            priority sw_police_id hw_policer_addr avgrate burst static_avgrate
avgrate_type
-------------------- -------- ------------ --------------- ------- ----- --------------
------------
unconfigured-default low      0            580096          500     100   500            2
UDP-default          low      1            580608          500     100   500            2
TCP-default          low      2            581120          500     100   500            2
Mcast-default        low      3            581632          500     100   500            2
Raw-listen           low      4            582144          500     100   500            2
Raw-default          low      5            582656          500     100   500            2
Fragment             low      6            583168          1000    100   1000           2
OSPF-mc-known        high     7            583680          2000    1000  2000           2
ISIS-known           high     8            584192          2000    1000  2000           2
EIGRP                high     9            584704          1500    750   1500           2
RIP                  high     10           585216          1500    750   1500           2
OSPF-mc-default      low      11           585728          1500    1000  1500           2
ISIS-default         low      12           586240          1500    1000  1500           2
BGP-known            high     13           586752          2500    1200  2500           2
BGP-cfg-peer         mdeium   14           587264          100     1000  2000           0
BGP-default          low      15           587776          100     750   1500           1
PIM-mcast-default    mdeium   16           588288          23000   100   23000          2
PIM-ucast            low      17           588800          10000   100   10000          2
IGMP                 mdeium   18           589312          3500    100   3500           2
ICMP-local           mdeium   19           589824          2500    100   2500           2
ICMP-app             low      20           590336          2500    100   2500           2
ICMP-default         low      21           590848          2500    100   2500           2
LDP-TCP-known        mdeium   22           591360          2500    1250  2500           2
LMP-TCP-known        mdeium   23           591872          2500    1250  2500           2
RSVP-UDP             mdeium   24           592384          7000    600   7000           2
RSVP-default         mdeium   25           592896          500     100   500            2
RSVP-known           mdeium   26           593408          7000    600   7000           2
IKE                  mdeium   27           593920          1000    100   1000           2
IPSEC-default        low      28           594432          1000    100   1000           2
IPSEC-known          mdeium   29           594944          3000    100   3000           2
MSDP-known           mdeium   30           595456          1000    100   1000           2
MSDP-cfg-peer        mdeium   31           595968          1000    100   1000           2
MSDP-default         low      32           596480          1000    100   1000           2
SNMP                 low      33           596992          2000    100   2000           2
NTP-default          high     34           597504          500     100   500            2
SSH-known            mdeium   35           598016          1000    100   1000           2
```

```
SSH-default          low      36          598528          1000    100   1000          2
HTTP-known           mdeium   37          599040          1000    100   1000          2
HTTP-default         low      38          599552          1000    100   1000          2
SHTTP-known          mdeium   39          600064          1000    100   1000          2
SHTTP-default        low      40          600576          1000    100   1000          2
TELNET-known         mdeium   41          601088          1000    100   1000          2
TELNET-default       low      42          601600          1000    100   1000          2
CSS-known            mdeium   43          602112          1000    100   1000          2
CSS-default          low      44          602624          1000    100   1000          2
RSH-known            mdeium   45          603136          1000    100   1000          2
RSH-default          low      46          603648          1000    100   1000          2
UDP-known            mdeium   47          604160          25000   100   25000         2
TCP-known            mdeium   48          604672          25000   100   25000         2
TCP-listen           low      49          605184          25000   100   25000         2
TCP-cfg-peer         mdeium   50          605696          25000   100   25000         2
Mcast-known          mdeium   51          606208          25000   100   25000         2
LDP-TCP-cfg-peer     mdeium   52          606720          2000    1000  2000          2
LMP-TCP-cfg-peer     mdeium   53          607232          2000    1000  2000          2
LDP-TCP-default      low      54          607744          1500    750   1500          2
LMP-TCP-default      low      55          608256          1500    750   1500          2
UDP-listen           low      56          608768          4000    100   4000          2
UDP-cfg-peer         mdeium   57          609280          4000    100   4000          2
LDP-UDP              mdeium   58          609792          2000    1000  2000          2
LMP-UDP              mdeium   59          610304          2000    1000  2000          2
All-routers          high     60          610816          1000    500   1000          2
OSPF-uc-known        high     61          611328          2000    1000  2000          2
OSPF-uc-default      low      62          611840          100     100   1000          0
ip-sla               high     63          612352          10000   100   10000         2
ICMP-control         high     64          612864          2500    100   2500          2
L2TPv3               mdeium   65          613376          25000   100   25000         2
PCEP                 mdeium   66          613888          100     200   100           2
GRE                  high     67          614400          1000    1000  1000          2
VRRP                 mdeium   68          614912          1000    1000  1000          2
HSRP                 mdeium   69          615424          400     400   400           2
BFD-known            critical 70          615936          8500    300   8500          2
BFD-default          critical 71          616448          8500    100   8500          2
MPLS-oam             mdeium   72          616960          100     100   100           2
DNS                  mdeium   73          617472          500     100   500           2
RADIUS               mdeium   74          617984          7000    600   7000          2
TACACS               mdeium   75          618496          500     100   500           2
PIM-mcast-known      mdeium   76          619008          23000   100   23000         2
BFD-MP-known         mdeium   77          619520          8400    1024  8400          2
BFD-MP-0             mdeium   78          620032          128     100   128           2
L2TPv2-default       mdeium   79          620544          700     100   700           2
NTP-known            high     80          621056          500     100   500           2
L2TPv2-known         mdeium   81          621568          2000    100   2000          2
```

The following table describes the significant fields shown in the display.

***Table 51: show lpts pifib hardware police Command Field Descriptions***

| Fleld | Description |
|---|---|
| FlowType | Type of flow that is binding between a tuple and a destination. |
| avgrate | Average policer rate in packets per second (PPS). |

## Related Commands

| Command | Description |
|---|---|
| flow (LPTS), on page 301 | Configures the policer for the LPTS flow type. |
| lpts pifib hardware police, on page 305 | Configures the ingress policers and enters pifib policer global configuration mode. |

# show lpts pifib statistics

To display Pre-Internal Forwarding Information Base (Pre-IFIB) statistics, use the **show lpts ifib statistics** command in XR EXEC mode.

**show lpts pifib statistics** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays Pre-IFIB statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts pifib statistics** command:

```
RP/0/RP0/CPU0:router# show lpts pifib statistics

Packets into Pre-IFIB:80
Lookups:80
Packets delivered locally:80
Packets delivered remotely:0
```

This table describes the significant fields shown in the display.

***Table 52: show lpts pifib statistics Command Field Descriptions***

| Field | Description |
|---|---|
| Packets into Pre-IFIB | Packets presented for pre-IFIB lookups. |
| Lookups | Packets looked up. |
| Packets delivered locally | Packets delivered to local applications or the local stack (*n* duplicated) packets duplicated for delivery to applications and the local stack. |
| Packets delivered remotely | Packets delivered to applications or for lookup on other RPs. |

## Related Commands

| Command | Description |
|---|---|
| show lpts pifib , on page 332 | Displays information about pre-IFIB entries. |

# show lpts port-arbitrator statistics

To display local packet transport services (LPTS) port arbitrator statistics, use the **show lpts port-arbitrator statistics** command in XR EXEC mode.

**show lpts port-arbitrator statistics**

## Syntax Description

This command has no keywords or arguments.

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| lpts | read |

## Examples

The following sample output is from the **show lpts port-arbitrator statistics** command:

```
RP/0/RP0/CPU0:router# show lpts port-arbitrator statistics

LPTS Port Arbitrator statistics:
 PA FGID-DB library statistics:
  0 FGIDs in use, 512 cached, 0 pending retries
  0 free allocation slots, 0 internal errors, 0 retry attempts
  1 FGID-DB notify callback, 0 FGID-DB errors returned
  FGID-DB permit mask: 0x7 (alloc mark rack0)
  PA API calls:
            1 init              1 realloc_done
            8 alloc             8 free
           16 join             16 leave
            8 detach
  FGID-DB API calls:
            1 register          1 clear_old
            1 alloc             0 free
           16 join             16 leave
            0 mark              1 mark_done
```

# show lpts punt excessive-flow-trap information

To display the Excessive Punt Flow Trap feature information, use the **show lpts punt excessive-flow-trap information** command in the XR EXEC mode.

**show lpts punt excessive-flow-trap information**

## Syntax Description

This command has no keywords or arguments.

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.2.2 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| lpts | read |
| basic-services | read, write |

This is an example of **show lpts punt excessive-flow-trap information** command with ARP protocol configured with default values:

```
RP/0/RP0/CPU0:router# show lpts punt excessive-flow-trap information

Sun Jun 13 12:42:06.122 UTC

------------------------------------------------------------
              Police          Penalty
              Rate (pps)      Timeout (mins)
 Protocol   Default Config   Default Config   Punt Reasons
 --------   --------------   --------------   ----------------
 ARP           10     -         15     -      ARP
                                              Reverse ARP

 ICMP           0     -         15     -

 DHCP           0     -         15     -

 PPPOE          0     -         15     -

 PPP            0     -         15     -

 IGMP           0     -         15     -

 IPv4/v6        0     -         15     -
```

```
L2TP            0      -          15      -

UNCLASSIFIED        0      -          15      -
```

The corresponding **show running-config** output for the above **show lpts punt excessive-flow-trap information** command is:

```
RP/0/RP0/CPU0:router# show running-config lpts punt excessive-flow-trap
lpts punt excessive-flow-trap
penalty-rate arp 10
penalty-timeout arp 15
non-subscriber-interfaces
!
```

This table describes the significant fields shown in the display.

***Table 53: show lpts punt excessive-flow-trap information Field Descriptions***

| Field | Description |
|---|---|
| penalty-rate | The penalty policing rate for a protocol. For ARP, the value is 10. |
| penalty-timeout | The penalty timeout value for a protocol. For ARP, the value is 15. |

## Related Commands

| Command | Description |
|---|---|
| show running-config lpts punt excessive-flow-trap, on page 357 | Displays the running configuration for the Excessive Punt Flow Trap feature. |

# show lpts punt excessive-flow-trap interface

To display the penalty status of an interface for one or all protocols, use the **show lpts punt excessive-flow-trap interface** command in the XR EXEC mode.

**show lpts punt excessive-flow-trap interface** *type interface-path-id* [**arp** ]

## Syntax Description

| | |
|---|---|
| *type* | Specifies the interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Either a physical interface instance or a virtual interface instance:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the modular services card or line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( ) and the module is CPU0. Example: interface MgmtEth0/ /CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| *arp* | Specifies the ARP protocol for which bad actors are displayed. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.2.2 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| lpts | read |
| basic-services | read, write |

The sample output for the **show lpts punt excessive-flow-trap arp** command is:

```
RP/0/RP0/CPU0:router# show lpts punt excessive-flow-trap arp
Wed Jun  4 15:54:54.087 PDT
    Parent Interface: TenGigE0/0/0/0/9                  Src MAC Addr: 0000.3357.ad6f

         Intf Handle: 0x00000058                            Location: 0/0/CPU0
            Protocol: ARP                                 Punt Reason: ARP
        Penalty Rate: 0 pps (all packets dropped)    Penalty Timeout: 5 mins

      Time Remaining: 3 mins 7 secs

    Parent Interface: TenGigE0/1/0/0                    Src MAC Addr: 0000.11e1.0a7a

         Intf Handle: 0x00800010                            Location: 0/1/CPU0
            Protocol: ARP                                 Punt Reason: ARP
        Penalty Rate: 0 pps (all packets dropped)    Penalty Timeout: 5 mins

      Time Remaining: 3 mins 7 secs
```

This table describes the significant fields shown in the display.

***Table 54: show lpts punt excessive-flow-trap interface Field Descriptions***

| Field | Description |
|---|---|
| Intf Handle | The interface handler for the Bundle Ether interface. |
| location | The location of the interface. |
| protocol | Specifies if it uses the IPv4 or IPv6 protocol. |
| punt reason | The reason to punt the excessive flow trap. |
| penalty-rate | The penalty policing rate for a protocol in pps. |
| penalty-timeout | The penalty timeout value for a protocol in minutes. |

## Related Commands

| Command | Description |
|---|---|
| show running-config lpts punt excessive-flow-trap, on page 357 | Displays the running configuration for the Excessive Punt Flow Trap feature. |

# show lpts punt excessive-flow-trap arp

To display a list of interfaces that are in the penalty box for ARP protocol, use the **show lpts punt excessive-flow-trap arp** command in the XR EXEC mode.

**show lpts punt excessive-flow-trap arp**

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.2.2 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operation |
|---|---|
| lpts | read |
| basic-services | read, write |

The sample output for the **show lpts punt excessive-flow-trap arp** command is:

```
RP/0/RP0/CPU0:router# show lpts punt excessive-flow-trap arp
Wed Jun  4 15:54:54.087 PDT
    Parent Interface: TenGigE0/0/0/0/9                 Src MAC Addr: 0000.3357.ad6f

         Intf Handle: 0x00000058                           Location: 0/0/CPU0
            Protocol: ARP                               Punt Reason: ARP
        Penalty Rate: 0 pps (all packets dropped)    Penalty Timeout: 5 mins

      Time Remaining: 3 mins 7 secs

    Parent Interface: TenGigE0/1/0/0                   Src MAC Addr: 0000.11e1.0a7a

         Intf Handle: 0x00800010                           Location: 0/1/CPU0
            Protocol: ARP                               Punt Reason: ARP
        Penalty Rate: 0 pps (all packets dropped)    Penalty Timeout: 5 mins

      Time Remaining: 3 mins 7 secs
```

This table describes the significant fields shown in the display.

***Table 55: show lpts punt excessive-flow-trap interface Field Descriptions***

| Field | Description |
|---|---|
| Intf Handle | The interface handler for the Bundle Ether interface. |

| Field | Description |
|---|---|
| location | The location of the interface. |
| protocol | Specifies the ARP protocol. |
| punt reason | The reason to punt the excessive flow trap. |
| penalty-rate | The penalty policing rate for a protocol in pps. |
| penalty-timeout | The penalty timeout value for a protocol in minutes. |

## Related Commands

| Command | Description |
|---|---|
| show running-config lpts punt excessive-flow-trap, on page 357 | Displays the running configuration for the Excessive Punt Flow Trap feature. |

# show running-config lpts punt excessive-flow-trap

To display the running configuration for the Excessive Punt Flow Trap feature, use the **show running-config lpts punt excessive-flow-trap** command in the XR EXEC mode.

**show running-configlpts punt excessive-flow-trap**

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.2.2 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operation |
|---|---|
| lpts | read |
| basic-services | read, write |

The **show running-config** output for the above **show lpts punt excessive-flow-trap** command is:

```
RP/0/RP0/CPU0:router# show running-config lpts punt excessive-flow-trap
lpts punt excessive-flow-trap
penalty-rate arp 10
penalty-timeout arp 20
non-subscriber-interfaces
!
```

This table describes the significant fields shown in the display.

***Table 56: show lpts punt excessive-flow-trap Field Descriptions***

| Field | Description |
|---|---|
| penalty-rate | The penalty policing rate for the ARP protocol. For ARP, the value is 10. |
| penalty-timeout | The penalty timeout value for the ARP protocol. For ARP, the value is 20. |

**Related Commands**

| Command | Description |
|---|---|
| lpts punt excessive-flow-trap, on page 307 | Enables the Excessive ARP Punt Protection feature. |
| show lpts punt excessive-flow-trap information, on page 351 | Displays the Excessive Punt Flow Trap information. |

| Command | Description |
| --- | --- |
| show lpts punt excessive-flow-trap interface, on page 353 | Displays the penalty status of an interface for one or all protocols. |

# Network Stack IPv4 and IPv6 Commands

This chapter describes the commands available on the Cisco IOS XR software to configure and monitor features related to IP Version 4 (IPv4) and IP Version 6 (IPv6).

For detailed information about network stack concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# clear ipv6 duplicate address

To trigger a Duplicate Address Detection (DAD) request for addresses that are found in DUPLICATE status, use the **clear ipv6 duplicate address** command. If a request is already triggered , then the **clear ipv6 duplicate address** command clears the DUPLICATE status of an address and makes it usable.

**clear ipv6 duplicate address** [*interface-type interface-path-id*] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *interface-type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | (Optional) Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) The designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If none of the optional keywords is specified, the command iterates through all the duplicate addresses and retriggers a DAD request for each of these addresses.

## Task ID

| Task ID | Operations |
|---|---|
| network | read, write |
| IPv6 | execute |

## Examples

The following example shows how to use the **clear ipv6 duplicate address** command:

```
RP/0/RP0/CPU0:router# clear ipv6 duplicate address
```

# clear ipv6 neighbors

To delete all entries in the IPv6 neighbor discovery cache, except static entries, use the **clear ipv6 neighbors** command in XR EXEC mode.

**clear ipv6 neighbors** [*interface-type interface-path-id*] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) The designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| *interface-type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | (Optional) Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If the location option is specified, only the neighbor entries specified in the **location** *node-id* keyword and argument are cleared.

## Task ID

| Task ID | Operations |
|---|---|
| network | read, write |
| IPv6 | execute |

## Examples

In the following example, only the highlighted entry is deleted:

```
RP/0/RP0/CPU0:router# show ipv6 neighbor

IPv6 Address Age Link-layer Addr State Interface
8888::3 - 1234.2345.9877 REACH HundredGigE0/0/0/0
8888::8 - 1234.2345.9877 REACH HundredGigE/0/0/0
```

```
fe80::205:1ff:fe9f:6400 1335 0005.019f.6400 STALE HundredGigE0/0/0/0
fe80::206:d6ff:fece:3808 1482 0006.d6ce.3808 STALE HundredGigE0/0/0/0
fe80::200:11ff:fe11:1112 1533 0000.1111.1112 STALE HundredGigE0/2/0/2

RP/0/RP0/CPU0:router# clear ipv6 neighbors location 0/2/0
RP/0/RP0/CPU0:router# show ipv6 neighbor

IPv6 Address Age Link-layer Addr State Interface
8888::3 - 1234.2345.9877 REACH HundredGigE0/0/0/0
8888::8 - 1234.2345.9877 REACH HundredGigE0/0/0/0
fe80::205:1ff:fe9f:6400 1387 0005.019f.6400 STALE HundredGigE0/0/0/0
fe80::206:d6ff:fece:3808 1534 0006.d6ce.3808 STALE HundredGigE0/0/0/0
```

# icmp ipv4 rate-limit unreachable

To limit the rate that IPv4 Internet Control Message Protocol (ICMP) destination unreachable messages are generated, use the **icmp ipv4 rate-limit unreachable** command in XR Config mode. To remove the rate limit, use the **no** form of this command.

**icmp ipv4 rate-limit unreachable** [{**DF** *milliseconds milliseconds*}][**disable**]
**no icmp ipv4 rate-limit unreachable** [{**DF** *milliseconds milliseconds*}][**disable**]

## Syntax Description

| | |
|---|---|
| **DF** | (Optional) Limits the rate at which ICMP destination unreachable messages are sent when code 4 fragmentation is needed and data fragmentation is (DF) set, as specified in the IP header of the ICMP destination unreachable message. |
| *milliseconds* | Time period (in milliseconds) between the sending of ICMP destination unreachable messages. Range is 1 to 4294967295. |
| **disable** | Disables ICMP destination unreachable message rate limiting. |

## Command Default

The default value is one ICMP destination unreachable message every 500 milliseconds.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The Cisco IOS XR softwaremaintains two timers: one for general destination unreachable messages and one for DF destination unreachable messages. Both share the same time limits and defaults. If the **DF** option is not configured, the **icmp ipv4 rate-limit unreachable** command sets the time values for DF destination unreachable messages. If the **DF** option is configured, its time values remain independent from those of general destination unreachable messages.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to set the time interval for the ICMP destination unreachable message to be generated at a minimum interval of 10 ms:

```
RP/0/RP0/CPU0:router(config)# icmp ipv4 rate-limit unreachable 10
```

# icmp source

To allow for flexible source IP address selection in the Internet Control Message Protocol (ICMP) response packet in response to a failure, use the **icmp source** command in the XR Config mode mode. To disallow flexible source IP address selection in the Internet Control Message Protocol (ICMP) response packet, use the **no** form of this command.

**icmp** [{**ipv4** | **ipv6**}] **source** [{**vrf** | **rfc**}]
**no icmp** [{**ipv4** | **ipv6**}] **source** [{**vrf** | **rfc**}]

## Syntax Description

| | |
|---|---|
| **ipv4** | IPv4 specific. |
| **ipv6** | IPv6 specific. |
| **source** | Enables source address selection policy. |
| **vrf** | Enable Strict VRF source address selection. |
| **rfc** | Enable RFC compliance for source address selection. |

## Command Default

None

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| network | read, write |

### Example

This example shows how to allow flexible source IP address corresponding to strict vrf in outgoing IPv6 ICMP packets.

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# icmp ipv6 source vrf
```

**Related Commands**

| Command | Description |
|---|---|
| icmp ipv4 rate-limit unreachable, on page 364 | Limits the rate that IPv4 Internet Control Message Protocol (ICMP) destination unreachable messages are generated. |

# ipv4 address (network)

To set a primary or secondary IPv4 address for an interface, use the **ipv4 address** command in interface configuration mode. To remove an IPv4 address, use the **no** form of this command.

**ipv4 address** *ipv4-address mask* [**secondary**] [**route-tag** *route-tag value*]
**no ipv4 address** *ipv4-address mask* [**secondary**] [**route-tag** *route-tag value*]

## Syntax Description

| | |
|---|---|
| **ipv4-address** | IPv4 address. |
| *mask* | Mask for the associated IP subnet. The network mask can be specified in either of two ways:<br>• The 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 belongs to the network address.<br>• The 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 network address. |
| **secondary** | (Optional) Specifies that the configured address is a secondary IPv4 address. If this keyword is omitted, the configured address is the primary IPv4 address. |
| **route-tag** | (Optional) Specifies that the configured address has a route tag to be associated with it. |
| *route-tag value* | (Optional) Value of the route tag. Range is 1 to 4294967295. |

## Command Default

No IPv4 address is defined for the interface.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

An interface can have one primary IPv4 address and multiple secondary IPv4 addresses. Packets generated by the software always use the primary IPv4 address. Therefore, all networking devices on a segment should share the same primary network number.

**Note** The same IPv4 address configured on two different interfaces causes an error message to display that indicates the conflict. The interface located in the highest rack, slot, module, instance, and port is disabled.

Hosts can determine subnet masks using the IPv4 Internet Control Message Protocol (ICMP) mask request message. Networking devices respond to this request with an ICMP mask reply message.

You can disable IPv4 processing on a particular interface by removing its IPv4 address with the **no ipv4 address** command. If the software detects another host using one of its IPv4 addresses, it will display an error message on the console.

The optional **secondary** keyword allows you to specify an unlimited number of secondary addresses. Secondary addresses are treated like primary addresses, except that the system never generates datagrams other than routing updates with secondary source addresses. IPv4 broadcasts and Address Resolution Protocol (ARP) requests are handled properly, as are interface routes in the IP routing table.

Secondary IPv4 addresses can be used in a variety of situations. The following are the most common applications:

- There may not be enough host addresses for a particular network segment. For example, your subnetting allows up to 254 hosts per logical subnet, but on one physical subnet you need to have 300 host addresses. Using secondary IPv4 addresses on the networking devices allows you to have two logical subnets using one physical subnet.
- Many older networks were built using Level 2 bridges. The judicious use of secondary addresses can aid in the transition to a subnetted, router-based network. Routers on an older, bridged segment can be easily made aware that there are many subnets on that segment.

The route-tag feature attaches a tag to all IPv4 addresses. The tag is propagated from the Management Agents (MA) to the Address Repository Managers (RPM) to routing protocols, thus enabling the user to control the redistribution of connected routes by looking at the route tags via RPL scripts.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to set 192.168.1.27 as the primary address and 192.168.7.17 and 192.168.8.17 as the secondary addresses on HundredGigE interface 0/1/1/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 192.168.1.27 255.255.255.0
RP/0/RP0/CPU0:router(config-if)# ipv4 address 192.168.7.17 255.255.255.0 secondary
RP/0/RP0/CPU0:router(config-if)# ipv4 address 192.168.8.17 255.255.255.0 secondary
```

## Related Commands

| Command | Description |
|---|---|
| show ipv4 interface , on page 432 | Lists a summary of IPv4 information and status for the interface. |

# ipv4 assembler max-packets

To configure the maximum number of packets that are allowed in assembly queues, use the **ipv4 assembler max-packets** command in XR Config mode. To disable this feature, use the **no** from of this command.

**ipv4 assembler max-packets** *percentage value*
**no ipv4 assembler max-packets** *percentage value*

## Syntax Description

| | |
|---|---|
| *percentage value* | Percentage of total packets available in the system. The range is from 1 to 50. |

## Command Default

None

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to configure the maximum number of packets for the assembly queue:

```
RP/0/RP0/CPU0:router(config)# ipv4 assembler max-packets 35
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 assembler timeout, on page 370 | Configures the number of seconds an assembly queue can hold before a timeout occurs. |

# ipv4 assembler timeout

To configure the number of seconds an assembly queue can hold before a timeout occurs, use the **ipv4 assembler timeout** command in XR Config mode. To disable this feature, use the **no** form of this command.

**ipv4 assembler timeout** *seconds*
**no ipv4 assembler timeout** *seconds*

## Syntax Description

| | |
|---|---|
| *seconds* | Number of seconds an assembly queue can hold before a timeout occurs. The range is from 1 to 120. |

## Command Default

None

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to configure an assembly queue before a timeout occurs:

```
RP/0/RP0/CPU0:router(config)# ipv4 assembler timeout 88
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 assembler max-packets, on page 369 | Configures the maximum number of packets that are allowed in assembly queues. |

# ipv4 conflict-policy

To enable IP Address Repository Manager (IPARM) conflict resolution, use the **ipv4 conflict-policy** command in XR Config mode. To disable the IPARM conflict resolution, use the **no** form of the command.

**ipv4 conflict-policy** {**highest-ip** | **longest-prefix** | **static**}
**no ipv4 conflict-policy** {**highest-ip** | **longest-prefix** | **static**}

## Syntax Description

| | |
|---|---|
| **highest-ip** | Keeps the highest ip address in the conflict set. |
| **longest-prefix** | Keeps the longest prefix match in the conflict set. |
| **static** | Keeps the existing interface running across new address configurations. |

## Command Default

The precedence rule adopted is loopback > physical > other virtual interfaces. Within virtual interfaces, there is an alphabetical preference, for example, loopback1 > loopback2 > tunnel. Among physical interfaces, the lower rack or slot takes control.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use **ipv4 conflict-policy** command to set an IPARM policy that resolves a conflict in the configured addresses. The policy tells IPARM what address to select from the addresses in conflict. The policy then forces the address in conflict to become inactive.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| ip-services | read, write |

## Examples

The following example shows how to enable the static policy for conflict resolution:

```
RP/0/RP0/CPU0:router(config)# ipv6 conflict-policy static
```

## Related Commands

| Command | Description |
|---|---|
| show arm conflicts, on page 420 | Displays the IPv4 or IPv6 address conflict information. |

# ipv4 directed-broadcast

To enable forwarding of IPv4 directed broadcasts on an interface, use the **ipv4 directed-broadcast** command in interface configuration mode. To disable forwarding of IPv4 directed broadcast on an interface, use the **no** form of this command.

**ipv4 directed-broadcast**
**no ipv4 directed-broadcast**

## Syntax Description

This command has no keywords or arguments.

## Command Default

By default, directed broadcasts are dropped.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

A directed broadcast is a packet sent to a specific network. IPv4 directed broadcasts are dropped and not forwarded. Dropping IPv4 directed broadcasts makes routers less susceptible to denial-of-service (DoS) attacks.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to enable the forwarding of IPv4 directed broadcasts on HundredGigE interface 0/1/1/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE 0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv4 directed-broadcast
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 unnumbered point-to-point | Enables IP processing on a point-to-point interface without assigning an explicit IP address to the interface. |
| show ipv4 interface , on page 432 | Lists a summary of IPv4 information and status for the interface. |

# ipv4 helper-address

To configure the address to which the software forwards User Datagram Protocol (UDP) broadcasts, including BOOTP, received on an interface, use the **ipv4 helper-address** command in interface configuration mode. To remove an IPv4 helper address, use the **no** form of this command.

**ipv4 helper-address** {*destination-address* | *destination-address*}
**no ipv4 helper-address** {*destination-address* | *destination-address*}

## Syntax Description

| | |
|---|---|
| *destination-address* | Destination broadcast or host address to be used when UDP broadcasts are forwarded. There can be more than one helper address per interface. |

## Command Default

IPv4 helper addresses are disabled. Default VRF is assumed if the VRF is not specified.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use this command with the **forward-protocol udp** command in XR Config mode, which specifies by port number the broadcast packets that are forwarded. UDP is enabled by default for well-known ports. The **ipv4 helper-address** command specifies the destination to which the UDP packets are forwarded.

One common application that requires IPv4 helper addresses is Dynamic Host Configuration Protocol (DHCP), which is defined in RFC 1531. DHCP protocol information is carried inside of BOOTP packets. To enable BOOTP broadcast forwarding for a set of clients, configure an IPv4 helper address on the networking device interface physically closest to the client. The IPv4 helper address should specify the address of the DHCP server. If you have multiple servers, you can configure one IPv4 helper address for each server. Because BOOTP packets are forwarded by default, DHCP information can now be forwarded by the networking device. The DHCP server now receives broadcasts from the DHCP clients.

A DHCP relay profile must be configured to perform DHCP Relay. The **ip helper-address** command is used to forward broadcast UDP (non-DHCP) packets.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to specify that all UDP broadcast packets received on HundredGigEinterface 0/1/1/0 are forwarded to 192.168.1.0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
```

```
RP/0/RP0/CPU0:router(config-if)# ipv4 helper-address 192.168.1.0
```

## Related Commands

| Command | Description |
|---|---|
| forward-protocol udp | Specifies which ports the networking device forwards to when forwarding broadcast packets. |

# ipv4 mask-reply

To enable the Cisco IOS XR softwareto respond to IPv4 Internet Control Message Protocol (ICMP) mask requests by sending ICMP mask reply messages, use the **ipv4 mask-reply** command in interface configuration mode. To restore the default, use the **no** form of this command.

**ipv4 mask-reply**
**no ipv4 mask-reply**

## Syntax Description

This command has no keywords or arguments.

## Command Default

IPv4 mask replies are not sent.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

This command enables the Cisco IOS XR softwareto respond to IPv4 ICMP mask requests by sending ICMP mask reply messages.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example enables the sending of ICMP mask reply messages on HundredGigEinterface 0/1/1/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv4 mask-reply
```

# ipv4 mtu

To set the maximum transmission unit (MTU) size of IPv4 packets sent on an interface, use the **ipv4 mtu** command in an appropriate configuration mode. To restore the default MTU size, use the **no** form of this command.

**ipv4 mtu** *bytes*

## Syntax Description

| | |
|---|---|
| *bytes* | MTU in bytes. Range is 68 to 65535 bytes for IPv4 packets. The maximum MTU size that can be set on an interface depends on the interface medium. |

## Command Default

If no MTU size is configured for IPv4 packets sent on an interface, the interface derives the MTU from the Layer 2 MTU.

## Command Modes

Interface configuration (for releases prior to R4.2.0)

Dynamic template configuration (for releases R4.2.0 onward)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The maximum MTU size that can be set on an interface depends on the interface medium. If the Layer 2 MTU is smaller than the Layer 3 MTU, the Cisco IOS XR software uses the Layer 2 MTU value for the Layer 3 MTU. Conversely, if the Layer 3 MTU is smaller than the Layer 2 MTU, the software uses Layer 3 MTU value. In other words the Cisco IOS XR software uses the lower of the two values for the MTU.

All devices on a physical medium must have the same protocol MTU to operate.

For releases R4.2.0 onward, to enter the dynamic template configuration mode, run the **dynamic-template** command in the XR Config mode.

**Note** Changing the MTU value (with the **mtu** interface configuration command) can affect the IPv4 MTU value. If the current IPv4 MTU value is the same as the MTU value, and you change the MTU value, the IPv4 MTU value will be modified automatically to match the new MTU. However, the reverse is not true; changing the IPv4 MTU value has no effect on the value for the **mtu** command.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

For releases prior to R4.2.0, this example shows how to set the maximum IPv4 packet size for HundredGigE interface 0/1/1/0 to 300 bytes:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv4 mtu 300
```

For releases R4.2.0 onward, this example shows how to set the maximum IPv4 packet size to 300 bytes in dynamic template configuration mode:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# dynamic-template type ppp p1
RP/0/RP0/CPU0:router(config-dynamic-template-type)# ipv4 mtu 300
```

## Related Commands

| Command | Description |
|---|---|
| show ipv4 interface , on page 432 | Displays the MTU status of interfaces configured for IPv4. |

# ipv4 redirects

To enable the sending of IPv4 Internet Control Message Protocol (ICMP) redirect messages if the software is forced to resend a packet through the same interface on which it was received, use the **ipv4 redirects** command in interface configuration mode. To restore the default, use the **no** form of this command.

**ipv4 redirects**
**no ipv4 redirects**

## Syntax Description

This command has no keywords or arguments.

## Command Default

ICMP redirect messages are disabled by default on the interface unless the Hot Standby Router Protocol (HSRP) is configured.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

ICMP redirect messages are disabled by default on the interface unless the Hot Standby Router Protocol (HSRP) is configured.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to disable the sending of ICMP IPv4 redirect messages on HundredGigE interface 0/1/1/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv4 redirects
```

# ipv4 source-route

To allow the processing of any IPv4 datagrams containing a source-route header option, use the **ipv4 source-route** command in XR Config mode. To have the software discard any IP datagram that contains a source-route option, use the **no** form of this command.

**ipv4 source-route**
**no ipv4 source-route**

## Syntax Description

This command has no keywords or arguments.

## Command Default

The software discards any IPv4 datagrams containing a source-route header option.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

By default, any IPv4 datagram which contains a source-route header option is discarded.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to allow the processing of any IPv4 datagrams containing a source-route header option:

```
RP/0/RP0/CPU0:router(config)# ipv4 source-route
```

# ipv4 unreachables disable

To disable the generation of IPv4 Internet Control Message Protocol (ICMP) unreachable messages, use the **ipv4 unreachables disable** command in an appropriate configuration mode. To re-enable the generation of ICMP unreachable messages, use the **no** form of this command.

**ipv4 unreachables disable**

## Syntax Description

This command has no keywords or arguments.

## Command Default

IPv4 ICMP unreachables messages are generated.

## Command Modes

Interface configuration (for releases prior to R4.2.0)

Dynamic template configuration (for releases R4.2.0 onward)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If the software receives a nonbroadcast packet destined for itself that uses a protocol it does not recognize, it sends an ICMP protocol unreachable message to the source.

If the software receives a datagram that it cannot deliver to its ultimate destination because it knows of no route to the destination address, it replies to the originator of that datagram with an ICMP host unreachable message.

This command affects a number of ICMP unreachable messages.

For releases R4.2.0 onward, to enter the dynamic template configuration mode, run the **dynamic-template** command in the XR Config mode.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

For releases prior to R4.2.0, this example shows how to disable the generation of ICMP unreachable messages on HundredGigEinterface 0/1/1/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv4 unreachables disable
```

For releases R4.2.0 onward, this example shows how to disable the generation of ICMP unreachable messages on dynamic template configuration mode:

```
RP/0/RP0/CPU0:router(config)# dynamic-template type ppp foo
RP/0/RP0/CPU0:router(config-dynamic-template-type)# ipv4 unreachables disable
```

# ipv4 virtual address

To define an IPv4 virtual address for a network of management Ethernet interfaces, use the **ipv4 virtual interface** command in XR Config mode. To remove an IPv4 virtual address from the configuration, use the **no** form of this command.

**ipv4 virtual address** {*ipv4-address/mask* | **use-as-src-addr**}
**no ipv4 virtual address** {*ipv4-address/mask* | **use-as-src-addr**}

## Syntax Description

| | |
|---|---|
| *ipv4 address* | Virtual IPv4 address and the mask that is to be unconfigured. |
| *mask* | Mask for the associated IP subnet. The network mask can be specified in either of two ways:<br>• The 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 belongs to the network address.<br>• The 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 network address. A slash between numbers is required as part of the notation. |
| **use-as-src-addr** | Enables the virtual address to be used as the default SRC address on sourced packets.<br>Cisco IOS XR Software Release 7.4.1 and later also supports virtual addresses for hosted Linux networking stack. |

## Command Default

No IPv4 virtual address is defined for the configuration.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 7.4.1 | This release supports virtual addresses for hosted Linux networking stack. |
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Configuring an IPv4 virtual address enables you to access the router from a single virtual address with a management network. An IPv4 virtual address persists across route processor (RP) failover situations.

Configuring an IPv4 virtual address enables you to access a dual RP router from a single address without prior knowledge of which RP is active. An IPv4 virtual address persists across RP failovers. For this to happen, the virtual IPv4 address must share a common IPv4 subnet with a Management Ethernet interface on both RPs.

The **use-as-src-addr** keyword eliminates the need for configuring a loopback interface as the source interface (that is, update source) for management applications. When an update source is not configured, management applications allow the transport processes (TCP, UDP, raw_ip) to pick a suitable source address. The transport processes, in turn, consult the FIB to do so. If a Management Ethernet's IP address is picked as the source address and if the **use-as-src-addr keyword** is configured, then the transport processes replace the Management Ethernet's IP address with a relevant virtual IP address. This functionality works across RP switchovers.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| network | read, write |

## Examples

The following example shows how to define an IPv4 virtual address:

```
RP/0/RP0/CPU0:router(config)# ipv4 virtual address 10.3.32.154/8
```

# ipv6 address

To configure an IPv6 address for an interface and enable IPv6 processing on the interface using an EUI-64 interface ID in the low-order 64 bits of the address, use the **ipv6 address** command in interface configuration mode. To remove the address from the interface, use the **no** form of this command.

**ipv6 address** *ipv6-prefix/prefix-length* [**eui-64**] [**route-tag** *route-tag value*]
**no ipv6 address** *ipv6-prefix/prefix-length* [**eui-64**] [**route-tag** *route-tag value*]

## Syntax Description

| | |
|---|---|
| *ipv6-prefix* | The IPv6 network assigned to the interface.<br>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* | The 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. |
| **eui-64** | (Optional) Specifies an interface ID in the low-order 64 bits of the IPv6 address. |
| **route-tag** | (Optional) Specifies that the configured address has a route tag to be associated with it. |
| *route-tag value* | (Optional) Value of the route tag. Range is 1 to 4294967295. |

## Command Default

No IPv6 address is defined for the interface.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If the value specified for the */ prefix-length* argument is greater than 64 bits, the prefix bits have precedence over the interface ID.

Using the **no ipv6 address** command without arguments removes all manually configured IPv6 addresses from an interface.

If the Cisco IOS XR software detects another host using one of its IPv6 addresses, it displays an error message on the console.

The route-tag feature attaches a tag to all IPv6 addresses. The tag is propagated from the Management Agents (MA) to the Address Repository Managers (RPM) to routing protocols, thus enabling the user to control the redistribution of connected routes by looking at the route tags via RPL scripts.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |

| Task ID | Operations |
|---|---|
| network | read, write |

## Examples

The following example assigns IPv6 address 2001:0DB8:0:1::/64 to HundredGigE interface 0/1/1/0 and specifies an EUI-64 interface ID in the low-order 64 bits of the address:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv6 address 2001:0DB8:0:1::/64 eui-64
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 address link-local, on page 386 | Configures an IPv6 link-local address for an interface and enables IPv6 processing on the interface. |
| show ipv6 interface , on page 439 | Displays the usability status of interfaces configured for IPv6. |

# ipv6 address link-local

To configure an IPv6 link-local address for an interface and enable IPv6 processing on the interface, use the **ipv6 address link-local** command in interface configuration mode. To remove the address from the interface, use the **no** form of this command.

**ipv6 address** *ipv6-address* **link-local** [**route-tag** *route-tag value*]
**no ipv6 address** *ipv6-address* **link-local** [**route-tag** *route-tag value*]

## Syntax Description

| | |
|---|---|
| *ipv6-address* | The IPv6 address assigned to the interface.<br>This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **link-local** | Specifies a link-local address. The *ipv6-address* value specified with this command overrides the link-local address that is automatically generated for the interface. |
| **route-tag** | (Optional) Specifies that the configured address has a route-tag to be associated with it. |
| *route-tag value* | (Optional) Displays the route-tag value. Range is 1 to 4294967295. |

## Command Default

No IPv6 address is defined for the interface.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If the Cisco IOS XR software detects another host using one of its IPv6 addresses, the software displays an error message on the console.

The system automatically generates a link-local address for an interface when IPv6 processing is enabled on the interface, typically when an IPv6 address is configured on the interface. To manually specify a link-local address to be used by an interface, use the **ipv6 address link-local** command.

A double colon may be used as part of the *ipv6-address* argument when consecutive 16-bit values are denoted as zero. You can configure multiple IPv6 addresses per interfaces, but only one link-local address.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |

## Examples

The following example shows how to assign FE80::260:3EFF:FE11:6770 as the link-local address for HundredGigE interface 0/1/1/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv6 address FE80::260:3EFF:FE11:6770 link-local
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 address, on page 384 | Configures an IPv6 address and enables IPv6 processing on an interface using an EUI-64 interface ID in the low-order 64 bits of the address. |
| show ipv6 interface , on page 439 | Displays the usability status of interfaces configured for IPv6. |

# ipv6 conflict-policy

To enable IP Address Repository Manager (IPARM) conflict resolution, use the **ipv6 conflict-policy** command in XR Config mode mode. To disable the IPARM conflict resolution, use the **no** form of the command.

**ipv6 conflict-policy** {**highest-ip** | **longest-prefix** | **static**}
**no ipv6 conflict-policy** {**highest-ip** | **longest-prefix** | **static**}

## Syntax Description

| | |
|---|---|
| **highest-ip** | Keeps the highest IP address in the conflict set. |
| **longest-prefix** | Keeps the longest prefix match in the conflict set. |
| **static** | Keeps the existing interface running across new address configurations. |

## Command Default

Default is the lowest rack/slot if no conflict policy is configured.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| ip-services | read, write |

## Examples

The following example shows how to enable the longest prefix policy for conflict resolution:

```
RP/0/RP0/CPU0:router(config)# ipv6 conflict-policy longest-prefix
```

# ipv6 enable

To enable IPv6 processing on an interface that has not been configured with an explicit IPv6 address, use the **ipv6 enable** command in an appropriate configuration mode. To disable IPv6 processing on an interface that has not been configured with an explicit IPv6 address, use the **no** form of this command.

**ipv6 enable**

## Syntax Description

This command has no keywords or arguments.

## Command Default

IPv6 is disabled.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **ipv6 enable** command automatically configures an IPv6 link-local unicast address on the interface while also enabling the interface for IPv6 processing. The **no ipv6 enable** command does not disable IPv6 processing on an interface that is configured with an explicit IPv6 address.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) shows how to enable IPv6 processing on HundredGigE interface 0/1/1/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv6 enable
```

For BNG, this example show how to enable IPv6 processing on dynamic template configuration mode:

```
RP/0/RP0/CPU0:router(config)# dynamic-template type ppp foo
RP/0/RP0/CPU0:router(config-dynamic-template-type)# ipv6 enable
```

**Related Commands**

| Command | Description |
|---|---|
| show ipv6 interface , on page 439 | Displays the usability status of interfaces configured for IPv6. |

# ipv6 hop-limit

To configure the maximum number of hops used in router advertisements and all IPv6 packets that are originated by the router, use the **ipv6 hop-limit** command in XR Config mode mode. To return the hop limit to its default value, use the **no** form of this command.

**ipv6 hop-limit** *hops*
**no ipv6 hop-limit** *hops*

## Syntax Description

| | |
|---|---|
| *hops* | Maximum number of hops. Range is 1 to 255. |

## Command Default

*hops* : 64 hops

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |

## Examples

The following example shows how to configure a maximum number of 15 hops for router advertisements and all IPv6 packets that are originated from the router:

```
RP/0/RP0/CPU0:router(config)# ipv6 hop-limit 15
```

# ipv6 icmp error-interval

To configure the interval and bucket size for IPv6 Internet Control Message Protocol (ICMP) error messages on all nodes, use the **ipv6 icmp error-interval** command in XR Config mode mode. To return the interval to its default setting, use the **no** form of this command.

**ipv6 icmp error-interval** *milliseconds* [*bucketsize*]
**no ipv6 icmp error-interval**

## Syntax Description

| | |
|---|---|
| *milliseconds* | Time interval (in milliseconds) between tokens being placed in the bucket. Range is 0 to 2147483647. |
| *bucketsize* | (Optional) The maximum number of tokens stored in the bucket. The acceptable range is 1 to 200 with a default of 10 tokens. |

## Command Default

ICMP rate limiting is enabled by default. To disable ICMP rate limiting, set the interval to zero.

*milliseconds* : 100 milliseconds

*bucketsize* : 10 tokens

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **ipv6 icmp error-interval** command in XR Config mode mode to limit the rate at which IPv6 ICMP error messages are sent for each node. A token bucket algorithm is used with one token representing one IPv6 ICMP error message. Tokens are placed in the virtual bucket at a specified interval until the maximum number of tokens allowed in the bucket is reached.

The *milliseconds* argument specifies the time interval between tokens being placed in the bucket. The optional *bucketsize* argument is used to define the maximum number of tokens stored in the bucket. Tokens are removed from the bucket when IPv6 ICMP error messages are sent, which means that if the *bucketsize* argument is set to 20, a rapid succession of 20 IPv6 ICMP error messages can be sent. When the bucket is empty of tokens, IPv6 ICMP error messages are not sent until a new token is placed in the bucket.

Use the **show ipv6 traffic** EXEC command to display IPv6 ICMP rate-limited counters.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |

## Examples

The following example shows an interval of 50 milliseconds and a bucket size of 20 tokens being configured for IPv6 ICMP error messages:

```
RP/0/RP0/CPU0:router(config)# ipv6 icmp error-interval 50 20
```

## Related Commands

| Command | Description |
|---|---|
| show ipv6 neighbors , on page 442 | Displays IPv6 neighbors discovery cache information. |

# ipv6 mtu

To set the maximum transmission unit (MTU) size of IPv6 packets sent on an interface, use the **ipv6 mtu** command in an appropriate configuration mode. To restore the default MTU size, use the **no** form of this command.

**ipv6 mtu** *bytes*

## Syntax Description

| | |
|---|---|
| *bytes* | MTU in bytes. Range is 1280 to 65535 for IPv6 packets. The maximum MTU size that can be set on an interface depends on the interface medium. |

## Command Default

If no MTU size is configured for IPv6 packets sent on an interface, the interface derives the MTU from the Layer 2 MTU.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If an IPv6 packet exceeds the MTU set for the interface, only the source router of the packet can fragment it.

The maximum MTU size that can be set on an interface depends on the interface medium. If the Layer 2 MTU is smaller than the Layer 3 MTU, the Cisco IOS XR software uses the Layer 2 MTU value for the Layer 3 MTU. Conversely, If the Layer 3 MTU is smaller than the Layer 2 MTU, the software uses Layer 3 MTU value. In other words the Cisco IOS XR software uses the lower of the two values for the MTU.

All devices on a physical medium must have the same protocol MTU to operate.

**Note** Changing the MTU value (with the **mtu** interface configuration command) can affect the IPv6 MTU value. If the current IPv6 MTU value is the same as the MTU value, and you change the MTU value, the IPv6 MTU value will be modified automatically to match the new MTU. However, the reverse is not true; changing the IPv6 MTU value has no effect on the value for the **mtu** command.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) shows how to set the maximum IPv6 packet size for HundredGigE interface 0/1/1/0 to 1350 bytes:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/1/0
RP/0/RP0/CPU0:router(config-if)# ipv6 mtu 1350
```

For BNG, this example shows how to set the maximum IPv6 packet size to 1350 bytes in the dynamic template configuration mode:

```
RP/0/RP0/CPU0:router(config)# dynamic-template type ppp foo
RP/0/RP0/CPU0:router(config-dynamic-template-type)# ipv6 mtu 1350
```

## Related Commands

| Command | Description |
|---|---|
| show ipv6 interface , on page 439 | Displays the usability status of interfaces configured for IPv6. |

# ipv6 nd

To configure Neighbor Discovery (ND) subcommands, use the **ipv6 nd** command in XR Config mode. To disable this feature, use the **no** form of this command.

**ipv6 nd** { **scavenge-timeout** *seconds* }

## Syntax Description

| | |
|---|---|
| **scavenge-timeout** *seconds* | Configures the lifetime of stale ipv6 neighb |

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

When the scavenge-timer for a neighbor entry expires, the entry is cleared.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |

## Examples

This example shows how to configure the timer to keep the neighbor in stale state in the cache:

```
Router(config)# ipv6 nd scavenge-timeout 3000
```

# ipv6 nd dad attempts

To configure the number of consecutive neighbor solicitation messages that are sent on an interface while duplicate address detection is performed on the unicast IPv6 addresses of the interface, use the **ipv6 nd dad attempts** command in an appropriate configuration mode. To return the number of messages to the default value, use the **no** form of this command.

**ipv6 nd dad attempts** *value*

## Syntax Description

| | |
|---|---|
| *value* | Number of neighbor solicitation messages. Range is 0 to 600. Configuring a value of 0 disables duplicate address detection processing on the specified interface; a value of 1 configures a single transmission without follow-up transmissions. |

## Command Default

Duplicate address detection on unicast IPv6 addresses with the sending of one neighbor solicitation message is enabled. The default is one message.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Duplicate address detection verifies the uniqueness of new unicast IPv6 addresses before the addresses are assigned to interfaces (the new addresses remain in a tentative state while duplicate address detection is performed). Duplicate address detection uses neighbor solicitation messages to verify the uniqueness of unicast IPv6 addresses.

The DupAddrDetectTransmits node configuration variable (as specified in RFC 2462, *IPv6 Stateless Address Autoconfiguration*) is used to automatically determine the number of consecutive neighbor solicitation messages that are sent on an interface while duplicate address detection is performed on a tentative unicast IPv6 address.

The interval between the sending of duplicate address detection neighbor solicitation messages (the duplicate address detection timeout interval) is specified by the neighbor discovery-related variable RetransTimer (as specified in RFC 2461, *Neighbor Discovery for IP Version 6 [IPv6]*), which is used to determine the time between retransmissions of neighbor solicitation messages to a neighbor when the address is being resolved or when the reachability of a neighbor is being probed. This is the same management variable used to specify the interval for neighbor solicitation messages during address resolution and neighbor unreachability detection. Use the **ipv6 nd ns-interval** command to configure the interval between neighbor solicitation messages that are sent during duplicate address detection.

Duplicate address detection is suspended on interfaces that are administratively down. While an interface is administratively down, the unicast IPv6 addresses assigned to the interface are set to a pending state. Duplicate address detection is automatically restarted on an interface when the interface returns to being administratively up.

**Note** An interface returning to administratively up restarts duplicate address detection for all of the unicast IPv6 addresses on the interface. While duplicate address detection is performed on the link-local address of an interface, the state for the other IPv6 addresses is still set to tentative. When duplicate address detection is completed on the link-local address, duplicate address detection is performed on the remaining IPv6 addresses.

When duplicate address detection identifies a duplicate address, the state of the address is set to duplicate and the address is not used. If the duplicate address is the link-local address of the interface, the processing of IPv6 packets is disabled on the interface and an error message similar to the following is issued:

```
ipv6_nd[145]: %IPV6_ND-3-ADDRESS_DUPLICATE : Duplicate address 111::1 has been detected
```

If the duplicate address is a global address of the interface, the address is not used and an error message similar to the following is issued:

```
%IPV6-4-DUPLICATE: Duplicate address 3000::4 on HundredGigE
```

All configuration commands associated with the duplicate address remain as configured while the state of the address is set to duplicate.

If the link-local address for an interface changes, duplicate address detection is performed on the new link-local address and all of the other IPv6 address associated with the interface are regenerated (duplicate address detection is performed only on the new link-local address).

Duplicate address detection is performed on all multicast-enabled IPv6 interfaces, including the following interface types:

- Cisco High-Level Data Link Control (HDLC)
- Ethernet, FastEthernet, and GigabitEthernet
- PPP

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) shows how to set the number of consecutive neighbor solicitation messages for interface 0/2/0/1 to 1 and then display the state (tentative or duplicate) of the unicast IPv6 address configured for an interface:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/2/0/1
RP/0/RP0/CPU0:router(config-if)# ipv6 nd dad attempts 1
RP/0/RP0/CPU0:router(config-if)# Uncommitted changes found, commit them before
exiting(yes/no/cancel)? [cancel]:y

RP/0/RP0/CPU0:router# show ipv6 interface
HundredGigE2/2/0/0 is Up, line protocol is Up
```

```
  IPv6 is disabled, link-local address unassigned
  No global unicast address is configured
HundredGigE2/2/0/1 is Up, line protocol is Up
  IPv6 is enabled, link-local address is fe80::203:fdff:fe1b:4501
  Global unicast address(es):
    1:4::1, subnet is 1:4::/64 [DUPLICATE]
  MTU is 1514 (1500 is available to IPv6)
  ICMP redirects are disabled
  ND DAD is enabled, number of DAD attempts 1
  ND reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
HundredGigE2/2/0/2 is Shutdown, line protocol is Down
  IPv6 is enabled, link-local address is fe80::200:11ff:fe11:1111 [TENTATIVE]
  Global unicast address(es):
    111::2, subnet is 111::/64 [TENTATIVE]
  MTU is 1514 (1500 is available to IPv6)
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts 1
  ND reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
```

For BNG, this example shows how to display the state (tentative or duplicate) of the unicast IPv6 address on the dynamic template configuration mode:

```
RP/0/RP0/CPU0:router(config)# dynamic-template type ppp p1
RP/0/RP0/CPU0:router(config-dynamic-template-type)# ipv6 nd dad attempts 1
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 nd ns-interval , on page 401 | Configures the interval between IPv6 neighbor solicitation transmissions on an interface. |

# ipv6 nd managed-config-flag

To set the managed address configuration flag in IPv6 router advertisements, use the **ipv6 nd managed-config-flag** command in an appropriate configuration mode. To clear the flag from IPv6 router advertisements, use the **no** form of this command.

**ipv6 nd managed-config-flag**

## Syntax Description

This command has no keywords or arguments.

## Command Default

The managed address configuration flag is not set in IPv6 router advertisements.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Setting the managed address configuration flag in IPv6 router advertisements indicates to attached hosts whether they should use stateful autoconfiguration to obtain addresses. If the flag is set, the attached hosts should use stateful autoconfiguration to obtain addresses. If the flag is not set, the attached hosts should not use stateful autoconfiguration to obtain addresses.

Hosts may use stateful and stateless address autoconfiguration simultaneously.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) shows how to configure the managed address configuration flag in IPv6 router advertisements on HundredGigE interface 0/1/1/0:

```
Router(config)# interface HundredGigE0/1/1/0
Router(config-if)# ipv6 nd managed-config-flag
```

For BNG, this example shows how to configure the managed address configuration flag in IPv6 router advertisements on dynamic template configuration mode:

```
Router(config)# dynamic-template type ppp p1
Router(config-dynamic-template-type)# ipv6 nd managed-config-flag
```

# ipv6 nd ns-interval

To configure the interval between IPv6 neighbor solicitation retransmissions on an interface, use the **ipv6 nd ns-interval** command in an appropriate configuration mode. To restore the default interval, use the **no** form of this command.

**ipv6 nd ns-interval** *milliseconds*

## Syntax Description

| | |
|---|---|
| *milliseconds* | Interval (in milliseconds) between IPv6 neighbor solicit transmissions. Range is 1000 to 4294967295. |

## Command Default

0 milliseconds (unspecified) is advertised in router advertisements, and the value 1000 is used for the neighbor discovery activity of the router itself.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

This value is included in all IPv6 router advertisements sent out from this interface. Very short intervals are not recommended in normal IPv6 operation. When a nondefault value is configured, the configured time is both advertised and used by the router itself.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) configures an IPv6 neighbor solicit transmission interval of 9000 milliseconds for HundredGigE interface 0/1/1/0:

```
Router(config)# interface HundredGigE0/1/1/0
Router(config-if)# ipv6 nd ns-interval 9000
```

For BNG, this example configures an IPv6 neighbor solicit transmission interval of 9000 milliseconds in the dynamic template configuration mode:

```
Router(config)# dynamic-template type ppp p1
Router(config-dynamic-template-type)# ipv6 nd ns-interval 9000
```

# ipv6 nd other-config-flag

To set the other stateful configuration flag in IPv6 router advertisements, use the **ipv6 nd other-config-flag** command in an appropriate configuration mode. To clear the flag from IPv6 router advertisements, use the **no** form of this command.

**ipv6 nd other-config-flag**

## Syntax Description

This command has no keywords or arguments.

## Command Default

The other stateful configuration flag is not set in IPv6 router advertisements.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The setting of the other stateful configuration flag in IPv6 router advertisements indicates to attached hosts how they can obtain autoconfiguration information other than addresses. If the flag is set, the attached hosts should use stateful autoconfiguration to obtain the other (nonaddress) information.

**Note** If the managed address configuration flag is set using the **ipv6 nd managed-config-flag** command, then an attached host can use stateful autoconfiguration to obtain the other (nonaddress) information regardless of the setting of the other stateful configuration flag.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) configures the “other stateful configuration” flag in IPv6 router advertisements on HundredGigE interface 0/1/1/0:

```
Router(config)# interface HundredGigE0/1/1/0
Router(config-if)# ipv6 nd other-config-flag
```

For BNG, this example configures the other stateful configuration flag for IPv6 router advertisements in the dynamic template configuration mode:

```
Router(config)# dynamic-template type ppp p1
Router(config-dynamic-template-type)# ipv6 nd other-config-flag
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 nd managed-config-flag , on page 400 | Sets the managed address configuration flag in IPv6 router advertisements. |

# ipv6 nd prefix

To configure how IPv6 prefixes are advertised in IPv6 router advertisements, use the **ipv6 nd prefix** command in interface configuration mode. To advertise a prefix with default parameter values, use the **no** form of this command. To prevent a prefix (or prefixes) from being advertised, use the **no-** keyword.

**ipv6 nd prefix** {*ipv6prefix*/*prefix-length* | **default** {*valid-lifetime* | **at** | **infinite** | **no-adv** | **no-autoconfig** | **off-link**}}
**no ipv6 nd prefix** {*ipv6prefix*/*prefix-length* | **default** {*valid-lifetime* | **at** | **infinite** | **no-adv** | **no-autoconfig** | **off-link**}}

## Syntax Description

| | |
|---|---|
| **ipv6-prefix** | The IPv6 network number to include in router advertisements.<br>This keyword must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| **/prefix-length** | The 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. |
| **default** | Specifies all prefixes. |
| **valid-lifetime** | The amount of time (in seconds) that the specified IPv6 prefix is advertised as being valid. The range of values is 0 to 4294967295 seconds. |
| **at** | The date and time at which the lifetime and preference expire. The prefix is valid until this specified date and time are reached. Dates are expressed in the form *date-valid-expire month-valid-expire hh:mm-valid-expire date-prefer-expire month-prefer-expire hh:mm-prefer-expire.* |
| **infinite** | The valid lifetime does not expire. |
| **no-adv** | The prefix is not advertised. |
| **no-autoconfig** | Indicates to hosts on the local link that the specified prefix cannot be used for IPv6 autoconfiguration. |
| **off-link** | Indicates that the specified prefix is assigned to the link. Nodes sending traffic to such addresses that contain the specified prefix consider the destination to be locally reachable on the link. This prefix should not be used for *onlink* determination. |

## Command Default

All prefixes configured on interfaces that originate IPv6 router advertisements are advertised with a valid lifetime of 2592000 seconds (30 days) and a preferred lifetime of 604800 seconds (7 days), and with both the "onlink" and "autoconfig" flags set.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

This command allows control over the individual parameters per prefix, including whether or not the prefix should be advertised.

To control how prefixes are advertised, use the **ipv6 nd prefix** command. By default, prefixes configured as addresses on an interface using the **ipv6 address** command are advertised with default values. If you configure prefixes for advertisement using the **ipv6 nd prefix** command, only the specified prefixes are advertised with the configured values, all other prefixes are advertised with default values.

The default keyword can be used to set default parameters for all prefixes.

A date can be set to specify the expiration of a prefix. The valid and preferred lifetimes are counted down in real time. When the expiration date is reached, the prefix is no longer advertised.

When onlink is "on" (by default), the specified prefix is assigned to the link. Nodes sending traffic to such addresses that contain the specified prefix consider the destination to be locally reachable on the link.

When autoconfig is "on" (by default), it indicates to hosts on the local link that the specified prefix can be used for IPv6 autoconfiguration.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |

## Examples

The following example includes the IPv6 prefix 2001:0DB8::/35 in router advertisements sent out HundredGigE interface 0/1/0/0 with a valid lifetime of 1000 seconds and a preferred lifetime of 900 seconds:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/1/0/0
RP/0/RP0/CPU0:router(config-if)# ipv6 nd prefix 2001:0DB8::/35 1000 900
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 address, on page 384 | Configures an IPv6 address and enables IPv6 processing on an interface using an EUI-64 interface ID in the low-order 64 bits of the address. |
| ipv6 address link-local, on page 386 | Configures an IPv6 link-local address for an interface and enables IPv6 processing on the interface. |
| ipv6 nd managed-config-flag , on page 400 | Sets the managed address configuration flag in IPv6 router advertisements. |
| show ipv6 interface , on page 439 | Displays the usability status of interfaces configured for IPv6. |

# ipv6 nd ra-interval

To configure the interval between IPv6 router advertisement transmissions on an interface, use the **ipv6 nd ra-interval** command in an appropriate configuration mode. To restore the default interval, use the **no** form of this command.

**ipv6 nd ra-interval** *maximum-interval* [*minimum-interval*]

## Syntax Description

| | |
|---|---|
| *maximum-interval* | Maximum router advertisement interval in seconds. |
| *minimum-interval* | Minimum router advertisement interval in seconds. |

## Command Default

*seconds* : 200 seconds

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The interval between transmissions should be less than or equal to the IPv6 router advertisement lifetime if the router is configured as a default router by using the **ipv6 nd ra-lifetime** command. To prevent synchronization with other IPv6 nodes, randomly adjust the actual value used to within 20 percent of the specified value.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) configures an IPv6 router advertisement interval of 201 seconds on HundredGigE interface 0/1/1/0:

```
Router(config)# interface HundredGigE0/1/1/0
Router(config-if)# ipv6 nd ra-interval 201
```

For BNG, this example configures an IPv6 router advertisement interval of 201 seconds in the dynamic template configuration mode:

```
Router(config)# dynamic-template type ppp p1Router
Router(config-dynamic-template-type)# ipv6 nd ra-interval 201
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 nd ra-lifetime , on page 408 | Configures the lifetime of an IPv6 router advertisement. |

# ipv6 nd ra-lifetime

To configure the router lifetime value in IPv6 router advertisements on an interface, use the **ipv6 nd ra-lifetime** command in an appropriate configuration mode. To restore the default lifetime, use the **no** form of this command.

**ipv6 nd ra-lifetime** *seconds*

## Syntax Description

| | |
|---|---|
| *seconds* | The validity (in seconds) of this router as a default router on this interface. |

## Command Default

*seconds* : 1800 seconds

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The router lifetime value is included in all IPv6 router advertisements sent out the interface. The value indicates the usefulness of the router as a default router on this interface. Setting the value to 0 indicates that the router should not be considered a default router on this interface. The router lifetime value can be set to a nonzero value to indicate that it should be considered a default router on this interface. The nonzero value for the router lifetime value should not be less than the router advertisement interval.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) configures an IPv6 router advertisement lifetime of 1801 seconds on HundredGigE interface 0/1/1/0:

```
Router(config)# interface HundredGigE0/1/1/0
Router(config-if)# ipv6 nd ra-lifetime 1801
```

For BNG, this example configures an IPv6 router advertisement lifetime of 1801 seconds in the dynamic template configuration mode:

```
Router(config)# dynamic-template type ppp p1
Router(config-dynamic-template-type)# ipv6 nd ra-lifetime 1801
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 nd ra-interval , on page 406 | Configures the interval between IPv6 router advertisement transmissions on an interface. |

# ipv6 nd reachable-time

To configure the amount of time that a remote IPv6 node is considered reachable after some reachability confirmation event has occurred, use the **ipv6 nd reachable-time** command in an appropriate configuration mode. To restore the default time, use the **no** form of this command.

**ipv6 nd reachable-time** *milliseconds*

## Syntax Description

| | |
|---|---|
| *milliseconds* | The amount of time (in milliseconds) that a remote IPv6 node is considered reachable. The range is from 0 to 3600000. |

## Command Default

0 milliseconds (unspecified) is advertised in router advertisements and 30000 (30 seconds) is used for the neighbor discovery activity of the router itself.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The configured time enables the router to detect unavailable neighbors. Shorter configured times enable the router to detect unavailable neighbors more quickly; however, shorter times consume more IPv6 network bandwidth and processing resources in all IPv6 network devices. Very short configured times are not recommended in normal IPv6 operation.

The configured time is included in all router advertisements sent out of an interface so that nodes on the same link use the same time value. A value of 0 indicates that the configured time is unspecified by this router.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) shows how to configure an IPv6 reachable time of 1,700,000 milliseconds for HundredGigE interface 0/1/1/0:

```
Router(config)# interface HundredGigE0/1/1/0
Router(config-if)# ipv6 nd reachable-time 1700000
```

For BNG, this example shows how to configure an IPv6 reachable time of 1,700,000 milliseconds in the dynamic template configuration mode:

```
Router(config)# dynamic-template type ppp p1
Router(config-dynamic-template-type)# ipv6 nd reachable-time 1700000
```

# ipv6 nd redirects

To send Internet Control Message Protocol (ICMP) redirect messages, use the **ipv6 nd redirects** command in interface configuration mode. To restore the system default, use the **no** form of this command.

**ipv6 nd redirects**
**no ipv6 nd redirects**

## Syntax Description

This command has no keywords or arguments.

## Command Default

The default value is disabled.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |

## Examples

The following example shows how to redirect IPv6 nd-directed broadcasts on HundredGigEinterface 0/2/0/2:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/0/0/0
0/2/0/2
RP/0/RP0/CPU0:router(config-if)# ipv6 nd redirects
```

## Related Commands

| Command | Description |
|---|---|
| show ipv6 interface , on page 439 | Displays the usability status of interfaces configured for IPv6. |

# ipv6 nd suppress-ra

To suppress IPv6 router advertisement transmissions on a LAN interface, use the **ipv6 nd suppress-ra** command in an appropriate configuration mode. To reenable the sending of IPv6 router advertisement transmissions on a LAN interface, use the **no** form of this command.

**ipv6 nd suppress-ra**

## Syntax Description

This command has no keywords or arguments.

## Command Default

IPv6 router advertisements are automatically sent on other types of interlaces if IPv6 unicast routing is enabled on the interfaces. IPv6 router advertisements are not sent on other types of interfaces.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **no ipv6 nd suppress-ra** command to enable the sending of IPv6 router advertisement transmissions on non-LAN interface types (for example, serial or tunnel interfaces).

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) shows how to suppress IPv6 router advertisements on HundredGigE interface 0/1/1/0:

```
Router(config)# interface HundredGigE0/1/1/0
Router(config-if)# ipv6 nd suppress-ra
```

For BNG, this example shows how to suppress IPv6 router advertisements in the dynamic template configuration mode:

```
Router(config)# dynamic-template type ppp p1
Router(config-dynamic-template-type)# ipv6 nd suppress-ra
```

For Cloud Native BNG, this example shows how to suppress IPv6 router advertisements in the cnbng-nal configuration mode:

# ipv6 neighbor

To configure a static entry in the IPv6 neighbor discovery cache, use the **ipv6 neighbor** command in XR Config mode mode. To remove a static IPv6 entry from the IPv6 neighbors discovery cache, use the **no** form of this command.

**ipv6 neighbor** *ipv6-address interface-type interface-instance hardware-address*
**no ipv6 neighbor** *ipv6-address interface-type interface-instance hardware-address*

## Syntax Description

| | |
|---|---|
| *ipv6-address* | The IPv6 address that corresponds to the local data-link address.<br>This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. |
| *interface-type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-instance* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the modular services card or line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| *hardware-address* | The local data-link address (a 48-bit address). |

## Command Default

Static entries are not configured in the IPv6 neighbor discovery cache.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **ipv6 neighbor** command is similar to the **arp** (global) command.

If an entry for the specified IPv6 address already exists in the neighbor discovery cache—learned through the IPv6 neighbor discovery process—the entry is automatically converted to a static entry.

Use the **show ipv6 neighbors** command to display static entries in the IPv6 neighbors discovery cache. A static entry in the IPv6 neighbor discovery cache has one state: reach (reachable)—The interface for this entry is up. If the interface for the entry is down, the **show ipv6 neighbors** command does not show the entry.

**Note** Reachability detection is not applied to static entries in the IPv6 neighbor discovery cache; therefore, the descriptions for the reach (reachable) state are different for dynamic and static cache entries. See the **show ipv6 neighbors** command for a description of the reach (reachable) state for dynamic cache entries.

The **clear ipv6 neighbors** command deletes all entries in the IPv6 neighbors discovery cache, except static entries. The **no ipv6 neighbor** command deletes a specified static entry from the neighbor discovery cache; the command does not remove dynamic entries—learned from the IPv6 neighbor discovery process—from the cache. Disabling IPv6 on an interface by using the **no ipv6 enable** or the **no ipv6 unnumbered** command deletes all IPv6 neighbor discovery cache entries configured for that interface, except static entries (the state of the entry changes to reach [reachable]).

Static entries in the IPv6 neighbor discovery cache are not modified by the neighbor discovery process.

**Note** Static entries for IPv6 neighbors can be configured only on IPv6-enabled LAN and ATM LAN Emulation interfaces.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |

## Examples

The following example shows how to configure a static entry in the IPv6 neighbor discovery cache for a neighbor with the IPv6 address 2001:0DB8::45A and link-layer address 0002.7D1A.9472 on ethernet interface 0/ RP0 /CPU0:

```
RP/0/RP0/CPU0:router(config)# ipv6 neighbor 2001:0DB8::45A 0002.7D1A.9472
```

## Related Commands

| Command | Description |
|---|---|
| clear ipv6 neighbors , on page 362 | Deletes all entries in the IPv6 neighbors discovery cache, except static entries. |
| ipv6 enable , on page 389 | Disables IPv6 processing on an interface that has not been configured with an explicit IPv6 address. |
| show ipv6 neighbors , on page 442 | Displays IPv6 neighbors discovery cache information. |

# ipv6 unreachables disable

To disable the generation of IPv6 Internet Control Message Protocol (ICMP) unreachable messages, use the **ipv6 unreachables disable** command in an appropriate configuration mode. To re-enable the generation of ICMP unreachable messages, use the **no** form of this command.

**ipv6 unreachables disable**

## Syntax Description

This command has no keywords or arguments.

## Command Default

IPv6 ICMP unreachables messages are generated.

## Command Modes

Interface configuration (not applicable for BNG)

Dynamic template configuration (for BNG)

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If the software receives a nonbroadcast packet destined for itself that uses a protocol it does not recognize, it sends an ICMP protocol unreachable message to the source.

If the software receives a datagram that it cannot deliver to its ultimate destination because it knows of no route to the destination address, it replies to the originator of that datagram with an ICMP host unreachable message.

This command affects a number of ICMP unreachable messages.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |
| config-services | read, write |

## Examples

This example (not applicable for BNG) shows how to disable the generation of ICMP unreachable messages on HundredGigE interface 0/6/0/0:

```
RP/0/RP0/CPU0:router(config)# interface HundredGigE0/6/0/0
RP/0/RP0/CPU0:router(config-if)# ipv6 unreachables disable
```

For BNG, this example shows how to disable the generation of ICMP unreachable messages on dynamic template configuration mode:

```
RP/0/RP0/CPU0:router(config)# dynamic-template type ppp foo
RP/0/RP0/CPU0:router(config-dynamic-template-type)# ipv6 unreachables disable
```

# ipv6 virtual address

To define an IPv6 virtual address for a network of management Ethernet interfaces, use the **ipv6 virtual address** command in XR Config mode. To remove an IPv6 virtual address from the configuration, use the **no** form of this command.

**ipv6 virtual address** {[**vrf** *vrf-name*]*ipv6-address/prefix-length* | **use-as-src-addr**}
**no ipv6 virtual address** {[**vrf** *vrf-name*]*ipv6-address/prefix-length* | **use-as-src-addr**}

## Syntax Description

| | |
|---|---|
| **vrf** *vrf-name* | (Optional) Configures the virtual address on a per VPN routing and forwarding (VRF) basis for the management interfaces The *vrf-name* argument specifies the name of the VRF. |
| *ipv6 address* | The virtual IPv6 address to be used. |
| *prefix-length* | The length of the IPv6 prefix. A decimal value that indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). A slash mark must precede the decimal value. |
| **use-as-src-addr** | Enables the virtual address to be used as the default SRC address on sourced packets.<br>Cisco IOS XR Software Release 7.4.1 and later also supports virtual addresses for hosted Linux networking stack. |

## Command Default

No IPv6 virtual address is defined for the configuration.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 7.4.1 | This release supports virtual addresses for hosted Linux networking stack. |
| Release 5.3.1 | This command was introduced. |

## Usage Guidelines

Configuring an IPv6 virtual address enables you to access the router from a single virtual address with a management network. An IPv6 virtual address persists across route processor (RP) failover situations.

Configuring an IPv6 virtual address enables you to access a dual RP router from a single address without prior knowledge of which RP is active. An IPv6 virtual address persists across RP failovers. For this to happen, the virtual IPv6 address must share a common IPv6 subnet with a Management Ethernet interface on both RPs.

If you disable the **ipv6 virtual address** command with the **vrf** keyword, the virtual IP address is unconfigured for the corresponding VRF or for the default if no VRF is specified. This results in the removal of the entry for the virtual IP address in the VRF table and in the ARP cache.

The default VRF is chosen when no VRF is specified. The virtual IP address is activated on a management interface that is attached to a default VRF.

The **use-as-src-addr** keyword eliminates the need for configuring a loopback interface as the source interface (that is, update source) for management applications. When an update source is not configured, management applications allow the transport processes (TCP, UDP, raw_ip) to pick a suitable source address. The transport

processes, in turn, consult the FIB to do so. If a Management Ethernet's IP address is picked as the source address and if the **use-as-src-addr keyword** is configured, then the transport processes replace the Management Ethernet's IP address with a relevant virtual IP address. This functionality works across RP switchovers.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| network | read, write |

## Examples

The following example shows how to define an IPv6 virtual address:

```
RP/0/RP0/CPU0:router(config)# ipv6 virtual address 0:0:0:7272::72/64
```

The following example shows how to configure the virtual IP addresses for management interfaces on a per VRF basis:

```
RP/0/RP0/CPU0:router(config)# ipv6 virtual address vrf ppp 0:0:0:7272::72/64
```

# show arm conflicts

To display IPv4 or IPv6 address conflict information identified by the Address Repository Manager (ARM), use the **show arm conflicts** command in XR EXEC mode.

**show arm** {**ipv4** | **ipv6**} **conflicts** [{**address** | **override** | **unnumbered**}]

## Syntax Description

| | |
|---|---|
| **ipv4** | Displays IPv4 address conflicts. |
| **ipv6** | Displays IPv6 address conflicts. |
| **address** | (Optional) Displays address conflict information. |
| **override** | (Optional) Displays address conflict override information. |
| **unnumbered** | (Optional) Displays unnumbered interface conflict information. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **show arm conflicts** command to display information about IPv4 or IPv6 address conflicts. You can use address conflict information to identify misconfigured IPv4 or IPv6 addresses.

Conflict information is displayed for interfaces that are forced down and for interfaces that are up.

Issuing the **show arm conflicts** command without specifying any optional keywords displays the output generated from both the **address** and **unnumbered** keywords.

## Task ID

| Task ID | Operations |
|---|---|
| network | read |

## Examples

The following sample output is from the **show arm ipv4 conflicts** command:

```
RP/0/RP0/CPU0:router# show arm ipv4 conflicts

F Forced down
| Down interface & addr                  Up interface & addr

F Lo2 10.1.1.2/24                         Lo1 10.1.1.1/24

Forced down interface          Up interface
tu2->tu1                       tu1->Lo1
```

The following is sample output from the **show arm ipv4 conflicts** command with the **address** keyword:

```
RP/0/RP0/CPU0:router# show arm ipv4 conflicts address

F Forced down
| Down interface & addr                  Up interface & addr

F Lo2 10.1.1.2/24                         Lo1 10.1.1.1/24
```

The following is sample output from the **show arm ipv4 conflicts** command with the **unnumbered** keyword:

```
RP/0/RP0/CPU0:router# show arm ipv4 conflicts unnumbered

Forced down interface          Up interface                  VRF
tu2->tu1                       tu1->Lo1
```

This table describes the significant fields shown in the display.

***Table 57: show arm conflicts Command Field Descriptions***

| Field | Description |
|---|---|
| Forced down | Legend defining a symbol that may appear in the output for this command. |
| Down interface & addr | Forced down interface name, type, and address. |
| Up interface & addr | List of interfaces that are up. |
| Forced down interface | Unnumbered interfaces that are in conflict and forced down. |
| Up interface | Unnumbered interfaces that are in conflict and are up. |

# show arm database

To display IPv4 or IPv6 address information stored in the Address Repository Manager (ARM) database, use the **show arm database** command in XR EXEC mode.

**show arm** {**ipv4** | **ipv6**} **database** [{**interface** *type interface-path-id* | **network** *prefix/length*}] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **ipv4** | Displays IPv4 address information. |
| **ipv6** | Displays IPv6 address information. |
| **interface** | Displays the IPv4 or IPv6 address configured on the specified interface. |
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **network** | Displays addresses that match a prefix. |
| *prefix / length* | Network prefix and mask. A slash (/) must precede the specified mask. The range is from 0 to 128. |
| **location** *node-id* | (Optional) The designated node. The node-id argument is entered in the rack/slot/module notation. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **show arm database** command should be used to display information in the IP ARM database. Database information is displayed with the IPv4 or IPv6 address, interface type and name, and producer information.

## Task ID

| Task ID | Operations |
|---|---|
| network | read |

## Examples

The following is sample output from the **show arm database** command:

```
RP/0/RP0/CPU0:router# show arm
 database
Fri Jul 25 10:54:52.304 PST DST

P = Primary, S = Secondary address
|U = Unnumbered
|| Address            Interface
Producer         Route-tag
VRF: default
P  172.29.52.75/24    MgmtEth0/RP0/CPU0/0           ipv4_ma 0/RP0/CPU0         100
P  10.2.2.2/32        Loopback0                     ipv4_ma 0/RP1/CPU0
P  10.12.24.2/24      Bundle-POS24                  ipv4_ma 0/RP1/CPU0
P  10.12.28.2/24      Bundle-Ether28                ipv4_ma 0/RP1/CPU0
P  10.12.29.2/24      Bundle-Ether28.1              ipv4_ma 0/RP1/CPU0
P  10.12.30.2/24      Bundle-Ether28.2              ipv4_ma 0/RP1/CPU0
P  10.12.31.2/24      Bundle-Ether28.3              ipv4_ma 0/RP1/CPU0
P
172.
29.
52.
76/24    MgmtEth0/RP1/CPU0/0  ipv4_ma 0/RP1/CPU0P  10.
112.
12.
2/24     TenGigE0/1/1/0 ipv4_ma 0/1/CPU0

| Address             Interface  Producer
P  10.12.16.2/24      GigabitEthernet0/1/5/0        ipv4_ma 0/1/CPU0           1001
P  10.23.4.2/24       GigabitEthernet0/1/5/1        ipv4_ma 0/1/CPU0           1002
P  10.27.4.2/24       GigabitEthernet0/1/5/2        ipv4_ma 0/1/CPU0
P  10.12.8.2/24       POS0/1/0/1                    ipv4_ma 0/1/CPU0
P  10.112.4.2/24      POS0/1/0/2                    ipv4_ma 0/1/CPU0
P  10.112.8.2/24      POS0/1/0/3                    ipv4_ma 0/1/CPU0
P  10.12.32.2/24      POS0/1/4/2                    ipv4_ma 0/1/CPU0
P  10.12.32.2/24      POS0/1/4/3                    ipv4_ma 0/1/CPU0
P  172.29.52.28/24    MgmtEth0/4/CPU1/0             ipv4_ma 0/4/CPU1
P  172.29.52.27/24    MgmtEth0/4/CPU0/0             ipv4_ma 0/4/CPU0
P  10.12.20.2/24      GigabitEthernet0/6/5/1        ipv4_ma 0/6/CPU0
P  10.
12.
40.
2/24 GigabitEthernet0/6/5/7 ipv4_ma 0/6/CPU0
S 10.4.2.4/24         gigabitethernet 10/0  ipv4_io 1 10
S 10.4.3.4/24         gigabitethernet 10/1  ipv4_io 1 10

P = Primary, S = Secondary address

|U = Unnumbered

|| Address            Interface                     Producer
VRF: default
P  10.12.12.2/24      POS0/6/0/1                    ipv4_ma 0/6/CPU0
P  10.23.8.2/24       POS0/6/4/4                    ipv4_ma 0/6/CPU0
P  10.12.4.2/24       POS0/6/4/5                    ipv4_ma 0/6/CPU0
P  10.24.4.2/24       POS0/6/4/6                    ipv4_ma 0/6/CPU0
P
10.27.
```

```
8.2/24POS0/6/4/7 ipv4_ma 0/6/CPU0
```

***Table 58: show arm database Command Field Descriptions***

| Field | Description |
|---|---|
| Primary | Primary IP address. |
| Secondary | Secondary IP address. |
| Unnumbered Address | Interface is unnumbered and the address displayed is that of the referenced interface. |
| Interface | Interface that has this IP address. |
| Producer | Process that provides the IP address to the ARM. |
| Route-tag | Route tag address. |

# show arm router-ids

To display the router identification information with virtual routing and forwarding table information for the Address Repository Manager (ARM), use the **show arm router-ids** command in XR EXEC mode.

**show arm** [**ipv4**] **router-ids**

**Syntax Description**

| | |
|---|---|
| **ipv4** | (Optional) Displays IPv4 router information. |

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

Use the **show arm router-ids** command with the **ipv4** keyword to display the selected router ID information for the router.

**Task ID**

| Task ID | Operations |
|---|---|
| network | read |

**Examples**

The following is sample output from the **show arm router-ids** command:

```
RP/0/RP0/CPU0:router# show arm router-ids

Router-ID         Interface

10.10.10.10       Loopback0
```

This table describes the significant fields shown in the display.

***Table 59: show arm router-ids Command Field Descriptions***

| Field | Description |
|---|---|
| Router-ID | Router identification. |
| Interface | Interface identification. |

# show arm registrations producers

To display producer registration information for the Address Repository Manager (ARM), use the **show arm registrations producers** command in XR EXEC mode.

**show arm {ipv4 | ipv6} registrations producers**

**Syntax Description**

| | |
|---|---|
| **ipv4** | Displays IPv4 producer registration information. |
| **ipv6** | Displays IPv6 producer registration information. |

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

Use the **show arm registrations producers** command to display information on producers of IP ARM registrations. Registration information is displayed with the ID.

**Task ID**

| Task ID | Operations |
|---|---|
| network | read |

**Examples**

The following is sample output from the **show arm registrations producers** command:

```
RP/0/RP0/CPU0:router# show arm ipv4 registrations producers

Id    Node            Producer Id   IPC Version Connected?
0     0/0/0           ipv4_io       1.1         Y
4     0/1/0           ipv4_io       1.1         Y
3     0/2/0           ipv4_io       1.1         Y
2     0/4/0           ipv4_io       1.1         Y
1     0/6/0           ipv4_io       1.1         Y
```

This table describes the significant fields shown in the display.

***Table 60: show arm registrations producers Command Field Descriptions***

| Field | Description |
|---|---|
| Id | An identifier used by the IP Address ARM (IP ARM) to keep track of the producer of the IP address. |
| Node | The physical node (RP/LC CPU) where the producer is running. |
| Producer Id | The string used by the producer when registering with IP ARM. |

| Field | Description |
|---|---|
| IPC Version | Version of the apis used by the producer to communicate with IP ARM. |
| Connected? | Status of whether the producer is connected or not. |

# show arm summary

To display summary information for the IP Address Repository Manager (ARM), use the **show arm summary** command in XR EXEC mode.

**show arm** {**ipv4** | **ipv6**} **summary** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **ipv4** | Displays IPv4 summary information. |
| **ipv6** | Displays IPv6 summary information. |
| **location** *node-id* | (Optional) The designated node. The node-id argument is entered in the rack/slot/module notation. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **show arm summary** command to display a summary of the number of producers, address conflicts, and unnumbered interface conflicts in the router.

## Task ID

| Task ID | Operations |
|---|---|
| network | read |

## Examples

The following is sample output from the **show arm summary** command:

```
Router# show arm ipv4 summary

IPv4 Producers                          :          3
IPv4 address conflicts                  :          0
IPv4 unnumbered interface conflicts     :          0
IPv4 DB Master version                  : 0x00000000
```

This table describes the significant fields shown in the display.

***Table 61: show arm summary Command Field Descriptions***

| Field | Description |
|---|---|
| IPv4 Producers | Number of IPv4 producers on the router. |
| IPv4 Router id consumers | Number of IPv4 router ID consumers on the router. |
| IPv4 address conflicts | Number of IPv4 address conflicts on the router. |

| Field | Description |
|---|---|
| IPv4 unnumbered interface conflicts | Number of IPv4 conflicts on unnumbered interfaces. |
| IPv4 DB Master version | IPv4 DB Master version |

# show clns statistics

To display Connectionless Network Service (CLNS) protocol statistics, use the **show clns statistics** command in XR EXEC mode.

**show clns statistics**

## Syntax Description

This command has no keywords or arguments.

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use this command to display CLNS statistics.

## Task ID

| Task ID | Operations |
|---|---|
| isis | read |

## Examples

The following is sample output from the **show clns statistics** command:

```
RP/0/RP0/CPU0:router# show clns statistics

CLNS Statistics:
Last counter clear:                    2868 seconds ago
Total number of packets sent:          0
Total number of packets received:      0
Send packets dropped, buffer overflow: 0
Send packets dropped, out of memory:   0
Send packets dropped, other:           0
Receive socket max queue size:         0
Class    Overflow/Max   Rate Limit/Max
IIH             0/0             0/0
LSP             0/0             0/0
SNP             0/0             0/0
OTHER           0/0             0/0
Total          0               0
```

This table describes the significant fields shown in the display.

***Table 62: show clns traffic Command Field Descriptions***

| Field | Description |
|---|---|
| Class | Indicates the packet type. Packets types are as follows:<br>• IIH—Intermediate System-to-Intermediate-System hello packets<br>• lsp—Link state packets<br>• snp—Sequence number packets<br>• other |
| Overflow/Max | Indicates the number of packet drops due to the socket queue being overflown. The count displays in an *x/y* format where *x* indicates the total number of packet drops and *y* indicates the maximum number of drops in a row. |
| Rate Limit/Max | Indicates the number of packet drops due to rate limitation. The count displays in an *x/y* format where *x* indicates the total number of packet drops and *y* indicates the maximum number of drops in a row. |

# show ipv4 interface

To display the usability status of interfaces configured for IPv4, use the **show ipv4 interface** command in the XR EXEC mode.

**show ipv4 interface** [{*type interface-path-id* | **brief** | **summary**}]

**Syntax Description**

| | |
|---|---|
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the modular services card or line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/ RP1 /CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **brief** | (Optional) Displays the primary IPv4 addresses configured on the router's interfaces and their protocol and line states. |
| **summary** | (Optional) Displays the number of interfaces on the router that are assigned, unassigned, or unnumbered. |

**Command Default**

If VRF is not specified, the software displays the default VRF.

**Command Modes**

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

The **show ipv4 interface** command provides output similar to the **show ipv6 interface** command, except that it is IPv4-specific.

**Task ID**

| Task ID | Operations |
|---|---|
| ipv4 | read |

| Task ID | Operations |
|---|---|
| network | read |

## Examples

This is the sample output of the **show ipv4 interface** command:

```
RP/0/RP0/CPU0:router# show ipv4 interface

Loopback0 is Up, line protocol is Up
  Internet address is
1.0.0.1/
8 with route-tag 110
  Secondary address 10.0.0.1/8
  MTU is 1514 (1514 is available to IP)
  Multicast reserved groups joined: 10.0.0.1
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound  access list is not set
  Proxy ARP is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
HundredGigE0/0/0/0 is Up, line protocol is Up
  Internet address is 10.25.58.1/16
  MTU is 1514 (1500 is available to IP)
  Multicast reserved groups joined: 224.0.0.1
  Directed broadcast forwarding is disabled
  Outgoing access list is not set
  Inbound  access list is not set
  Proxy ARP is enabled
  ICMP redirects are always sent
  ICMP unreachables are always sent
HundredGigE0/0/0/0 is Shutdown, line protocol is Down

  Internet protocol processing disabled
```

This table describes the significant fields shown in the display.

***Table 63: show ipv4 interface Command Field Descriptions***

| Field | Description |
|---|---|
| Loopback0 is Up | If the interface hardware is usable, the interface is marked “Up.” For an interface to be usable, both the interface hardware and line protocol must be up. |
| line protocol is Up | If the interface can provide two-way communication, the line protocol is marked “Up.” For an interface to be usable, both the interface hardware and line protocol must be up. |
| Internet address | IPv4 Internet address and subnet mask of the interface. |
| Secondary address | Displays a secondary address, if one has been set. |
| MTU | Displays the IPv4 MTU[9] value set on the interface. |
| Multicast reserved groups joined | Indicates the multicast groups this interface belongs to. |

| Field | Description |
|---|---|
| Directed broadcast forwarding | Indicates whether directed broadcast forwarding is enabled or disabled. |
| Outgoing access list | Indicates whether the interface has an outgoing access list set. |
| Inbound access list | Indicates whether the interface has an incoming access list set. |
| Proxy ARP | Indicates whether proxy ARP[10] is enabled or disabled on an interface. |
| ICMP redirects | Specifies whether ICMPv4[11] redirects are sent on this interface. |
| ICMP unreachables | Specifies whether unreachable messages are sent on this interface. |
| Internet protocol processing disabled | Indicates an IPv4 address has not been configured on the interface. |

[9] MTU = maximum transmission unit
[10] ARP = Address Resolution Protocoladdress resolution protocol
[11] ICMPv4 = Internet Control Message Protocol internet control message protocol version 4

# show kim status

The Kernel Interface Module (KIM) is an IOS XR process that ensures IOS XR and Linux have consistent views of the required network state such as interfaces, routes, VRFs and so on.

To display the status of KIM, use the **show kim status** command in the XR EXEC mode. KIM is used to trigger the creation of route, interface, vrf and so on in the kernel. KIM also handles the programming of Local Packet Transport Services (LPTS) in response to the events that applications use to open sockets (TCP, UDP) in the kernel.

**show kim status vrf default**

## Syntax Description

| | |
|---|---|
| **vrf default** | Displays the default status of Virtual Private Network (VPN) routing and forwarding (VRF) reference. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 7.4.1 | Extended support for virtual IP addresses. |
| Release 6.0 | This command was introduced. |

## Usage Guidelines

Only the default VRF is supported.

## Task ID

| Task ID | Operations |
|---|---|
| system | read |

## Examples

This is the sample output of the **show kim status vrf default** command:

```
RP/0/RP0/CPU0:router# show kim status vrf default
Features:
  VRF namespaces                : Enabled
  VLAN interfaces               : Enabled
  VRF dataport interfaces       : Disabled
IM Connection                  : Connected (1 attempts/0 disconnects)
LPTS PA Connection             : Connected (0 disconnects)
Num socket bindings            : 0
Num Interfaces                 : 56
Loopback interfaces            : 1
Mgmt interfaces                : 1
LC interfaces                  : 54
IPv4 RIB routes                : 0
IPv6 RIB routes                : 0
Forwarding LC NPU ID           : 144
```

```
Forwarding i/f MTU            : 1482
IPV4 Source Address           : via Default selection
                     Interface: Loopback999
               Chosen source IP: 9.9.9.9
IPV6 Source Address           : via Default selection
                     Interface: Loopback999
               Chosen source IP: 999:999::9
IPV4 Virtual Address          : 1.2.3.4/24
IPV6 Virtual Address          : None
```

# show ipv4 traffic

To display the IPv4 traffic statistics, use the **show ipv4 traffic** command in the XR EXEC mode.

**show ipv4 traffic** [**brief**]

**Syntax Description**

| | |
|---|---|
| **brief** | (Optional) Displays only IPv4 and Internet Control Message Protocol version 4 (ICMPv4) traffic. |

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

The **show ipv4 traffic** command provides output similar to the **show ipv6 traffic** command, except that it is IPv4-specific.

**Task ID**

| Task ID | Operations |
|---|---|
| ipv4 | read |
| network | read |

**Examples**

This is the sample output of the **show ipv4 traffic** command:

```
RP/0/RP0/CPU0:router# show ipv4 traffic

IP statistics:
  Rcvd:  16372 total, 16372 local destination
         0 format errors, 0 bad hop count
         0 unknown protocol, 0 not a gateway
         0 security failures, 0 bad source, 0 bad header
         0 with options, 0 bad, 0 unknown
  Opts:  0 end, 0 nop, 0 basic security, 0 extended security
         0 strict source rt, 0 loose source rt, 0 record rt
         0 stream ID, 0 timestamp, 0 alert, 0 cipso
  Frags: 0 reassembled, 0 timeouts, 0 couldn't reassemble
         0 fragmented, 0 fragment count
  Bcast: 0 sent, 0 received
  Mcast: 0 sent, 0 received
   Drop: 0 encapsulation failed, 0 no route, 0 too big, 0 sanity address check
   Sent: 16372 total

ICMP statistics:
  Sent: 0 admin unreachable, 0 network unreachable
        0 host unreachable, 0 protocol unreachable
        0 port unreachable, 0 fragment unreachable
        0 time to live exceeded, 0 reassembly ttl exceeded
        5 echo request, 0 echo reply
        0 mask request, 0 mask reply
```

```
        0 parameter error, 0 redirects
        5 total
  Rcvd: 0 admin unreachable, 0 network unreachable
        2 host unreachable, 0 protocol unreachable
        0 port unreachable, 0 fragment unreachable
        0 time to live exceeded, 0 reassembly ttl exceeded
        0 echo request, 5 echo reply
        0 mask request, 0 mask reply
        0 redirect, 0 parameter error
        0 source quench, 0 timestamp, 0 timestamp reply
        0 router advertisement, 0 router solicitation
        7 total, 0 checksum errors, 0 unknown

UDP statistics:
        16365 packets input, 16367 packets output
        0 checksum errors, 0 no port
        0 forwarded broadcasts

TCP statistics:
        0 packets input, 0 packets output
        0 checksum errors, 0 no port
```

This table describes the significant fields shown in the display.

***Table 64: show ipv4 traffic Command Field Descriptions***

| Field | Description |
|---|---|
| bad hop count | Occurs when a packet is discarded because its TTL[12] field was decremented to zero. |
| encapsulation failed | Usually indicates that the router had no ARP request entry and therefore did not send a datagram. |
| format errors | Indicates a gross error in the packet format, such as an impossible Internet header length. |
| IP statistics Rcvd total | Indicates the total number of local destination and other packets received in the software plane. It does not account for the IP packets forwarded or discarded in hardware. |
| no route | Counted when the Cisco IOS XR software discards a datagram it did not know how to route. |

[12] TTL = time-to-live

# show ipv6 interface

To display the usability status of interfaces configured for IPv6, use the **show ipv6 interface** command in the XR EXEC mode.

**show ipv6 interface** [{*type interface-path-id* | **brief** | **summary**}]

**Syntax Description**

| | |
|---|---|
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the modular services card or line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/ RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **brief** | (Optional) Displays the primary IPv6 addresses configured on the router interfaces and their protocol and line states. |
| **link-local** | (Optional) Displays the link local IPv6 address. |
| **global** | (Optional) Displays the global IPv6 address. |
| **summary** | (Optional) Displays the number of interfaces on the router that are assigned, unassigned, or unnumbered. |

**Command Default**

None

**Command Modes**

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

The **show ipv6 interface** command provides output similar to the **show ipv4 interface** command, except that it is IPv6-specific.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read |

## Examples

This is the sample output of the **show ipv6 interface** command:

```
RP/0/RP0/CPU0:router# show ipv6 interface

HundredGigE0/2/0/0 is Up, line protocol is Up,
  IPv6 is enabled, link-local address is fe80::212:daff:fe62:c150
  Global unicast address(es):
    202::1, subnet is 202::/64
  Joined group address(es): ff02::1:ff00:1 ff02::1:ff62:c150 ff02::2
      ff02::1
  MTU is 1514 (1500 is available to IPv6)
  ICMP redirects are disabled
  ICMP unreachables are enabled
  ND DAD is enabled, number of DAD attempts 1
  ND reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  Hosts use stateless autoconfig for addresses.
  Outgoing access list is not set
  Inbound  access list is not set
```

This table describes the significant fields shown in the display.

***Table 65: show ipv6 interface Command Field Descriptions***

| Field | Description |
|---|---|
| HundredGigE/3/0/0 is Shutdown, line protocol is Down | Indicates whether the interface hardware is currently active (whether line signal is present) and whether it has been taken down by an administrator. If the interface hardware is usable, the interface is marked “Up.” For an interface to be usable, both the interface hardware and line protocol must be up. |
| line protocol is Up (or down) | Indicates whether the software processes that handle the line protocol consider the line usable (that is, whether keepalives are successful). If the interface can provide two-way communication, the line protocol is marked “Up.” For an interface to be usable, both the interface hardware and line protocol must be up. |
| IPv6 is enabled, stalled, disabled (stalled and disabled are not shown in sample output) | Indicates that IPv6 is enabled, stalled, or disabled on the interface. If IPv6 is enabled, the interface is marked “enabled.” If duplicate address detection processing identified the link-local address of the interface as being a duplicate address, the processing of IPv6 packets is disabled on the interface and the interface is marked “stalled.” If IPv6 is not enabled, the interface is marked “disabled.” |
| link-local address | Displays the link-local address assigned to the interface. |

| Field | Description |
|---|---|
| TENTATIVE | The state of the address in relation to duplicate address detection. States can be any of the following:<br>• duplicate—The address is not unique and is not being used. If the duplicate address is the link-local address of an interface, the processing of IPv6 packets is disabled on that interface.<br>• tentative—Duplicate address detection is either pending or under way on this interface.<br>**Note** If an address does not have one of these states (the state for the address is blank), the address is unique and is being used. |
| Global unicast addresses | Displays the global unicast addresses assigned to the interface. |
| ICMP redirects | State of Internet Control Message Protocol (ICMP) IPv6 redirect messages on the interface (the sending of the messages is enabled or disabled). |
| ND DAD | State of duplicate address detection on the interface (enabled or disabled). |
| number of DAD attempts | Number of consecutive neighbor solicitation messages that are sent on the interface while duplicate address detection is performed. |
| ND reachable time | Displays the neighbor discovery reachable time (in milliseconds) assigned to this interface. |

## Related Commands

| Command | Description |
|---|---|
| show ipv4 interface , on page 432 | Displays the usability status of interfaces configured for IPv4. |

# show ipv6 neighbors

To display the IPv6 neighbor discovery cache information, use the **show ipv6 neighbors** command in the XR EXEC mode.

**show ipv6 neighbors** [{*type interface-path-id* | **location** *node-id* | *ipv6prefix*/*prefix-length*}]

## Syntax Description

| | |
|---|---|
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface instance or a virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **location** *node-id* | (Optional) Designates a node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| *ipv6prefix* | (Optional) The IPv6 network assigned to the interface.<br>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* | The 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. |

## Command Default

All IPv6 neighbor discovery cache information is displayed.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

When the *interface-type* and *interface-number* arguments are not specified, cache information for all IPv6 neighbors is displayed. Specifying the *interface-type* and *interface-number* arguments displays only cache information about the specified interface.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read |

## Examples

This is the sample output of the **show ipv6 neighbors** command when entered with an interface type and number:

```
RP/0/RP0/CPU0:router# show ipv6 neighbors HundredGigE0/0/0/0

IPv6 Address                              Age Link-layer Addr State Interface
2000:0:0:4::2                               0 0003.a0d6.141e  REACH HundredGigE2
FE80::203:A0FF:FED6:141E                    0 0003.a0d6.141e  REACH HundredGigE2
3001:1::45a                                 - 0002.7d1a.9472  REACH HundredGigE2
```

This is the sample output of the **show ipv6 neighbors** command when entered with an IPv6 address:

```
RP/0/RP0/CPU0:router# show ipv6 neighbors 2000:0:0:4::2

IPv6 Address                              Age Link-layer Addr State Interface
2000:0:0:4::2                               0 0003.a0d6.141e  REACH HundredGigE2
```

This table describes significant fields shown in the display.

***Table 66: show ipv6 neighbors Command Field Descriptions***

| Field | Description |
|---|---|
| IPv6 Address | IPv6 address of neighbor or interface. |
| Age | Time (in minutes) since the address was confirmed to be reachable. A hyphen (-) indicates a static entry. |
| Link-layer Addr | MAC address. If the address is unknown, a hyphen (-) is displayed. |

| Field | Description |
|---|---|
| State | The state of the neighbor cache entry. These are the states for dynamic entries in the IPv6 neighbor discovery cache:<br>• INCMP (incomplete)—Address resolution is being performed on the entry. A neighbor solicitation message has been sent to the solicited-node multicast address of the target, but the corresponding neighbor advertisement message has not yet been received.<br>• reach (reachable)—Positive confirmation was received within the last ReachableTime milliseconds that the forward path to the neighbor was functioning properly. While in reach state, the device takes no special action as packets are sent.<br>• stale—More than ReachableTime milliseconds have elapsed since the last positive confirmation was received that the forward path was functioning properly. While in stale state, the device takes no action until a packet is sent.<br>• delay—More than ReachableTime milliseconds have elapsed since the last positive confirmation was received that the forward path was functioning properly. A packet was sent within the last DELAY_FIRST_PROBE_TIME seconds. If no reachability confirmation is received within DELAY_FIRST_PROBE_TIME seconds of entering the delay state, send a neighbor solicitation message and change the state to probe.<br>• probe—A reachability confirmation is actively sought by resending neighbor solicitation messages every RetransTimer milliseconds until a reachability confirmation is received.<br>These are the possible states for static entries in the IPv6 neighbor discovery cache:<br>• reach (reachable)—The interface for this entry is up.<br>• INCMP (incomplete)—The interface for this entry is down.<br>**Note** Reachability detection is not applied to static entries in the IPv6 neighbor discovery cache; therefore, the descriptions for the INCMP (incomplete) and reach (reachable) states are different for dynamic and static cache entries. |
| Interface | Interface from which the address is reachable. |

## Related Commands

| Command | Description |
|---|---|
| show ipv6 neighbors summary , on page 445 | Displays summary information for the neighbor entries. |

# show ipv6 neighbors summary

To display summary information for the neighbor entries, use the **show ipv6 neighbors summary** command in the XR EXEC mode.

**show ipv6 neighbors summary**

## Syntax Description

None

## Command Default

The default value is disabled.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read |

## Examples

This is the sample output of the **show ipv6 neighbors summary** command that shows the summary information for the neighbor entries:

```
RP/0/RP0/CPU0:router# show ipv6 neighbors summary

Mcast nbr entries:
    Subtotal: 0
Static nbr entries:
    Subtotal: 0
Dynamic nbr entries:
    Subtotal: 0

Total nbr entries: 0
```

## Related Commands

| Command | Description |
|---|---|
| show ipv6 neighbors , on page 442 | Displays IPv6 neighbor discovery cache information. |

# show ipv6 traffic

To display the IPv6 traffic statistics, use the **show traffic** command in the XR EXEC mode.

**show ipv6 traffic** [**brief**]

**Syntax Description**

| | |
|---|---|
| **brief** | (Optional) Displays only IPv6 and Internet Control Message Protocol version 6 (ICMPv6) traffic statistics. |

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

The **show ipv6 traffic** command provides output similar to the **show ipv4 traffic** command, except that it is IPv6-specific.

**Task ID**

| Task ID | Operations |
|---|---|
| ipv6 | read |
| network | read |

**Examples**

This is the sample output of the **show ipv6 traffic** command:

```
RP/0/RP0/CPU0:router# show ipv6 traffic

 IPv6 statistics:
  Rcvd:  0 total, 0 local destination
         0 source-routed, 0 truncated
         0 format errors, 0 hop count exceeded
         0 bad header, 0 unknown option, 0 bad source
         0 unknown protocol
         0 fragments, 0 total reassembled
         0 reassembly timeouts, 0 reassembly failures
         0 reassembly max drop
         0  sanity address check drops
  Sent:  0 generated, 0 forwarded
         0 fragmented into 0 fragments, 0 failed
         0 no route, 0 too big
  Mcast: 0 received, 0 sent

ICMP statistics:
  Rcvd: 0 input, 0 checksum errors, 0 too short
        0 unknown error type
        unreach: 0 routing, 0 admin, 0 neighbor,
                 0 address, 0 port, 0 unknown
        parameter: 0 error, 0 header, 0 option,
                   0 unknown
```

```
        0 hopcount expired, 0 reassembly timeout,
        0 unknown timeout, 0 too big,
        0 echo request, 0 echo reply
  Sent: 0 output, 0 rate-limited
        unreach: 0 routing, 0 admin, 0 neighbor,
                 0 address, 0 port, 0 unknown
        parameter: 0 error, 0 header, 0 option
                 0 unknown
        0 hopcount expired, 0 reassembly timeout,
        0 unknown timeout, 0 too big,
        0 echo request, 0 echo reply

Neighbor Discovery ICMP statistics:
  Rcvd: 0 router solicit, 0 router advert, 0 redirect
        0 neighbor solicit, 0 neighbor advert
  Sent: 0 router solicit, 0 router advert, 0 redirect
        0 neighbor solicit, 0 neighbor advert

UDP statistics:
        0 packets input, 0 checksum errors
        0 length errors, 0 no port, 0 dropped
        0 packets output

TCP statistics:s
        0 packets input, 0 checksum errors, 0 dropped
        0 packets output, 0 retransmitted
```

This table describes the significant fields shown in the display.

***Table 67: show ipv6 traffic Command Field Descriptions***

| Field | Description |
|---|---|
| Rcvd: | Statistics in this section refer to packets received by the router. |
| total | Total number of packets received by the software. |
| local destination | Locally destined packets received by the software. |
| source-routed | Packets seen by the software with RH. |
| truncated | Truncated packets seen by the software. |
| bad header | An error was found in generic HBH, RH, DH, or HA. Software only. |
| unknown option | Unknown option type in IPv6 header. |
| unknown protocol | Protocol specified in the IP header of the received packet is unreachable. |
| Sent: | Statistics in this section refer to packets sent by the router. |
| forwarded | Packets forwarded by the software. If the packet cannot be forwarded in the first lookup (for example, the packet needs option processing), then the packet is not included in this count, even if it ends up being forwarded by the software. |
| Mcast: | Multicast packets. |
| ICMP statistics: | Internet Control Message Protocol statistics. |

**Related Commands**

| Command | Description |
|---|---|
| show ipv4 traffic , on page 437 | Displays statistics about IPv4 traffic. |

# show mpa client

To display information about the Multicast Port Arbitrator (MPA) clients, use the **show mpa client** command in XR EXEC mode.

**show mpa client** {**consumers** | **producers**}

## Syntax Description

| | |
|---|---|
| **consumers** | Displays the clients for the consumers. |
| **producers** | Displays the clients for the producers. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| network | read |

## Examples

The following sample output is from the **show mpa client** command:

```
RP/0/RP0/CPU0:router# show mpa client producers

List of producer clients for ipv4 MPA

Location     Protocol     Process

0/RP1/CPU0   17           udp
0/RP1/CPU0   255          raw
```

***Table 68: show mpa client Command Field Descriptions***

| Field | Description |
|---|---|
| List of producer clients for MPA | Displays the producer clients that have registered with MPA. |
| Location | Displays the node on which the producer client is hosted. |
| Protocol | Displays the IP protocol ID. |
| Process | Displays the name of the producer client. |

# show mpa groups

To display Multicast Port Arbitrator (MPA) multicast group information, use the **show mpa groups** command in XR EXEC mode.

**show mpa groups** *type interface-path-id*

**Syntax Description**

| | |
|---|---|
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the modular services card or line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/ RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

**Command Default**

None

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operations |
|---|---|
| network | read |

**Examples**

The following sample output is from the **show mpa groups** command:

```
RP/0/RP0/CPU0:router# show mpa groups gig 0/1/0/2
Mon Jul 27 04:07:19.802 DST
HundredGigE0/1/0/2 :-
```

```
224.0.0.1 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
224.0.0.2 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
224.0.0.5 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
224.0.0.6 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
224.0.0.13 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
224.0.0.22 : includes 0, excludes 1, mode EXCLUDE
  <no source filter>
```

*Table 69: show mpa groups Command Field Descriptions*

| Field | Description |
|---|---|
| Includes | Displays the number of client registrations that have enabled the group in the include mode. |
| Excludes | Displays the number of client registrations that have enabled the group in the exclude mode. |
| Mode | Displays the current mode for the address. |
| No source filter | Indicates that the router does not have the desired list of IP addresses. |

**Note** The source filter consists of a list of source IP addresses. Depending on the mode, the list identifies the set of addresses from where multicast packets are either allowed or disallowed. In the include mode, the router accepts packets only from the IP addresses that are present in the source filter. In the exclude mode, the router drops packets from addresses that are present in the source filter. No source filter indicates that the registration does not have such a filter.

# show mpa ipv4

To display information for Multicast Port Arbitrator (MPA) for IPv4, use the **show mpa ipv4** command in XR EXEC mode.

**show mpa ipv4** {**client** {**consumers** | **producers**} | **groups** *type* *interface-path-id* | **trace**}

## Syntax Description

| | |
|---|---|
| **client** | Displays information about the MPA clients. |
| **consumers** | Displays the clients for the consumers. |
| **producers** | Displays the clients for the producers. |
| **groups** | Displays information about the MPA multicast group. |
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the modular services card or line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **trace** | Displays MPA trace information |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| network | read |

## Examples

The following sample output is from the **show mpa ipv4** command:

```
RP/0/RP0/CPU0:router# show mpa ipv4 client producers

List of producer clients for ipv4 MPA

 Location     Protocol     Process

 0/RP1/CPU0   255          raw
 0/RP1/CPU0   17           udp
```

***Table 70: show mpa ipv4 Command Field Descriptions***

| Field | Description |
|---|---|
| List of producer clients for ipv4 MPA | Displays the producer clients that have registered with MPA. |
| Location | Displays the node on which the producer client is hosted. |
| Protocol | Displays the IP protocol ID. |
| Process | Displays the name of the producer client. |

# show mpa ipv6

To display information for Multicast Port Arbitrator (MPA) for IPv6, use the **show mpa ipv6** command in XR EXEC mode.

**show mpa ipv6** {**client** {**consumers** | **producers**} | **groups** *type* *interface-path-id* | **trace**}

## Syntax Description

| | |
|---|---|
| **client** | Displays information about the MPA clients. |
| **consumers** | Displays the clients for the consumers. |
| **producers** | Displays the clients for the producers. |
| **groups** | Displays information about the MPA multicast group. |
| **type** | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>• *rack*: Chassis number of the rack.<br>• *slot*: Physical slot number of the modular services card or line card.<br>• *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>• *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric ( RP0 or RP1) and the module is CPU0. Example: interface MgmtEth0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **trace** | Displays MPA trace information |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| network | read |

## Examples

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

```
RP/0/RP0/CPU0:router# show mpa ipv6 client producers

List of producer clients for ipv6 MPA

 Location     Protocol     Process

 0/RP1/CPU0   17           udp
 0/RP1/CPU0   255          raw
```

*Table 71: show mpa ipv6 Command Field Descriptions*

| Field | Description |
|---|---|
| List of producer clients for ipv6 MPA | Displays the producer clients that have registered with MPA. |
| Location | Displays the node on which the producer client is hosted. |
| Protocol | Displays the IP protocol ID. |
| Process | Displays the name of the producer client. |

# Prefix List Commands

This chapter describes the Cisco IOS XR software commands used to configure IP Version 4 (IPv4) and IP Version 6 (IPv6) prefix lists.

For detailed information about prefix list concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# clear prefix-list ipv4

To reset the hit count on an IP Version 4 (IPv4) prefix list, use the **clear prefix-list ipv4** command in XR EXEC mode.

**clear prefix-list ipv4** *name* [*sequence-number*]

## Syntax Description

| | |
|---|---|
| *name* | Name of the prefix list from which the hit count is to be cleared. |
| *sequence-number* | (Optional) Sequence number of a prefix list. Range is 1 to 2147483646. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The hit count is a value indicating the number of matches to a specific prefix list entry. Use the **clear prefix-list ipv4** command to clear counters for a specified configured prefix list.

Use the *sequence-number* argument to clear counters for a prefix list with a specific sequence number.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

The following example displays IPv4 prefix lists, shows how to clear the counters for list3, then shows how to display the IPv4 prefix lists again, showing that counters are cleared for list3:

```
RP/0/RP0/CPU0:router# show prefix-list ipv4

ipv4 prefix-list list1
 10 permit 172.18.30.154/16 (8 matches)
ipv4 prefix-list list2
 20 deny 172.24.30.164/16 (12 matches)
ipv4 prefix-list list3
 30 permit 172.19.31.154/16 (32 matches)

RP/0/RP0/CPU0:router# clear prefix-list ipv4 list3

RP/0/RP0/CPU0:router# show prefix-list ipv4

ipv4 prefix-list list1
 10 permit 172.18.30.154/16 (8 matches)
ipv4 prefix-list list2
 20 deny 172.24.30.164/16 (12 matches)
```

```
ipv4 prefix-list list3
 30 permit 172.19.31.154/16
```

## Related Commands

| Command | Description |
|---|---|
| deny (prefix-list), on page 466 | Sets deny conditions for an IPv4 or IP IPv6 prefix list. |
| ipv4 prefix-list, on page 469 | Defines an IPv4 prefix list. |
| permit (prefix-list), on page 473 | Sets permit conditions for an IPv4 or IPv6 prefix list. |
| show prefix-list ipv4, on page 484 | Displays the configuration of the current IPv4 prefix list. |

# clear prefix-list ipv6

To reset the hit count on an IP Version 6 (IPv6) prefix list, use the **clear prefix-list ipv6** command in XR EXEC mode.

**clear prefix-list ipv6** *name* [*sequence-number*]

## Syntax Description

| | |
|---|---|
| *name* | Name of the prefix list from which the hit count is to be cleared. |
| *sequence-number* | (Optional) Clears counters for a prefix list with a specific sequence number. Range is 1 to 2147483646. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The hit count is a value indicating the number of matches to a specific prefix list entry. Use the **clear prefix-list ipv6** command to clear counters for a specified configured prefix list.

Use the *sequence-number* argument to clear counters for a prefix list with a specific sequence number.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

The following example shows IPv6 prefix lists, clears the counters for sequence number 60 on prefix list list3, then displays the IPv6 prefix lists again, showing that counters are cleared for sequence number 60:

```
RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list1
 40 permit 2000:1::/64 (5 matches)
 60 deny 3000:1::/64 (7 matches)

RP/0/RP0/CPU0:router# clear prefix-list ipv6 list1 60
RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list1
 40 permit 2000:1::/64 (5 matches)
 60 deny 3000:1::/64
```

## Related Commands

| Command | Description |
|---|---|
| deny (prefix-list), on page 466 | Sets deny conditions for an IPv4 or IPv6 prefix list. |
| ipv6 prefix-list, on page 471 | Defines an IPv6 prefix list. |
| permit (prefix-list), on page 473 | Sets permit conditions for an IPv4 or IPv6 prefix list. |
| show prefix-list ipv6, on page 487 | Displays the contents of the current IPv6 prefix list. |

# copy prefix-list ipv4

To create a copy of an existing IP Version 4 (IPv4) prefix list, use the **copy prefix-list ipv4** command in XR EXEC mode.

**copy prefix-list ipv4** *source-name destination-name*

## Syntax Description

| | |
|---|---|
| *source-name* | Name of the prefix list to be copied. |
| *destination-name* | Destination prefix list where the contents of the *source-name* will be copied. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **copy prefix-list ipv4** command to copy a configured prefix list. Use the *source-name* argument to specify the prefix list to be copied and the *destination-name* argument to specify where to copy the contents of the source prefix list. The *destination-name* argument must be a unique name; if the *destination-name* argument name exists for a prefix list or access list, the prefix list is not copied. The **copy prefix-list ipv4** command checks that the source prefix list exists, then checks the existing list names to prevent overwriting existing prefix lists.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| filesystem | execute |

## Examples

The following example displays IPv4 prefix lists, shows how to copy prefix-list1 to list4, then displays the IPv4 prefix lists again, showing prefix list4:

```
RP/0/RP0/CPU0:router# show prefix-list ipv4

ipv4 prefix-list list1
 10 permit 172.24.20.164/16
ipv4 prefix-list list2
 20 deny 172.18.30.154/16
ipv4 prefix-list list3
 30 permit 172.29.30.154/16

RP/0/RP0/CPU0:router# copy prefix-list ipv4 list1 list4

RP/0/RP0/CPU0:router# show prefix-list ipv4
ipv4 prefix-list list1
 10 permit 172.24.20.164/16
```

```
ipv4 prefix-list list2
 20 deny 172.18.30.154/16
ipv4 prefix-list list3
 30 permit 172.29.30.154/16
ipv4 prefix-list list4
 10 permit 172.24.20.164/16
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 prefix-list, on page 469 | Defines an IPv4 prefix list. |
| show prefix-list ipv4, on page 484 | Displays the contents of the current IPv4 prefix lists. |

# copy prefix-list ipv6

To create a copy of an existing IP Version 6 (IPv6) prefix list, use the **copy prefix-list ipv6** command in XR EXEC mode.

**copy prefix-list ipv6** *source-name destination-name*

## Syntax Description

| | |
|---|---|
| *source-name* | Name of the prefix list to be copied. |
| *destination-name* | Destination prefix list where the contents of the *source-name* will be copied. |

## Command Default

No default behavior or values

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **copy prefix-list ipv6** command to copy a configured prefix list. Use the *source-name* argument to specify the prefix list to be copied and the *destination-name* argument to specify where to copy the contents of the source prefix list. The *destination-name* argument must be a unique name; if the *destination-name* argument name exists for a prefix list or access list, the prefix list is not copied. The **copy prefix-list ipv6** command checks that the source prefix list exists then checks the existing list names to prevent overwriting existing prefix lists.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| filesystem | execute |

## Examples

The following example shows IPv6 prefix lists, shows how to copy prefix-list1 to list4, then displays the IPv6 prefix lists again, showing prefix list4:

```
RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list1
 40 permit 2000:1::/64
 60 deny 3000:1::/64
ipv6 prefix-list list2
 10 permit 5555::/24

RP/0/RP0/CPU0:router# copy prefix-list ipv6 list1 list3

RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list1
 40 permit 2000:1::/64
 60 deny 3000:1::/64
ipv6 prefix-list list2
```

```
 10 permit 5555::/24
ipv6 prefix-list list3
 40 permit 2000:1::/64
 60 deny 3000:1::/6
```

## Related Commands

| Command | Description |
|---|---|
| ipv6 prefix-list, on page 471 | Defines an IPv6 prefix list. |
| show prefix-list ipv6, on page 487 | Displays the contents of current IPv6 prefix list. |

# deny (prefix-list)

To set deny conditions for an IP Version 4 (IPv4) or IP Version 6 (IPv6) prefix list, use the **deny** command in IPv4 prefix list configuration or IPv6 prefix list configuration modes. To remove a condition from a prefix list, use the **no** form of this command.

*[sequence-number]* **deny** *network/length* [**ge** *value*] [**le** *value*] [**eq** *value*]
**no** *sequence-number* **deny**

## Syntax Description

| | |
|---|---|
| *sequence-number* | (Optional) Sets deny conditions for a prefix list with a specific sequence number. If you do not use a sequence number, the condition defaults to the next available sequence number in the prefix list. Range is 1 to 2147483646. By default, the first statement is number 10, and the subsequent statements are incremented by 10. The **sequence-number** argument must be used with the **no** form of the command. |
| *network / length* | Network number and length (in bits) of the network mask. |
| **ge** *value* | (Optional) Specifies a prefix length greater than or equal to the value. It is the lowest value of a range of the *length* (the "from" portion of the length range). |
| **le** *value* | (Optional) Specifies a prefix length less than or equal to the value. It is the highest value of a range of the *length* (the "to" portion of the length range). |
| **eq** *value* | (Optional) Exact value of the *length* . |

## Command Default

There is no specific condition under which a packet is denied passing the IPv4 or IPv6 prefix list.

## Command Modes

IPv4 prefix list configuration

IPv6 prefix list configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **deny** command to specify conditions under which a packet cannot pass the prefix list.

The **ge**, **le** and **eq** keywords can be used to specify the range of the prefix length to be matched, for prefixes that are more specific than the *network/length* argument. Exact match is assumed when neither **ge** nor **le** is specified. The range is assumed to be from the **ge** *value* to 32 if only the **ge** keyword is specified. The range is assumed to be from the *length* to the **le** *value argument* if only the **le** attribute is specified.

A specified **ge** *value* or **le** *value* must satisfy the following condition:

*length* < **ge** *value* < **le** *value* <= 32 (for IPv4)

*length* < **ge** *value* < **le** *value* <= 128 (for IPv6)

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

The following example shows how to deny the route 10.0.0.0/0:

```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list list1
RP/0/RP0/CPU0:router(config-ipv4_pfx)# 50 deny 10.0.0.0/0
```

The following example shows how to deny all routes with a prefix of 10.3.32.154:

```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list list1
RP/0/RP0/CPU0:router(config-ipv4_pfx)#80 deny 10.3.32.154 le 32
```

The following example shows how to deny all masks with a length greater than 25 bits routes with a prefix of 172.18.30.154/16:

```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list list1
RP/0/RP0/CPU0:router(config-ipv4_pfx)#100 deny 172.18.30.154/16 ge 25
```

The following example shows how to deny mask lengths greater than 25 bits in all address space:

```
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list list2
RP/0/RP0/CPU0:router(config-ipv6_pfx)# 70 deny  2000:1::/64 ge 25
```

The following example shows how to add deny conditions to list3, then use the **no** form of the command to remove the condition with the sequence number 30:

```
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list list3

RP/0/RP0/CPU0:router(config-ipv6_pfx)# deny 2000:1::/64 ge 25
RP/0/RP0/CPU0:router(config-ipv6_pfx)# deny 3000:1::/64 le 32
RP/0/RP0/CPU0:router(config-ipv6_pfx)# deny 4000:1::/64 ge 25
Uncommitted changes found, commit them? [yes]: y

RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list3
 10 deny 2000:1::/64 ge 25
 20 deny 3000:1::/64 le 32
 30 deny 4000:1::/64 ge 25

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list list3
RP/0/RP0/CPU0:router(config-ipv6_pfx)# no 30
Uncommitted changes found, commit them? [yes]: y
RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list3
 10 deny 2000:1::/64 ge 25
```

```
20 deny 3000:1::/64 le 32
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 prefix-list, on page 469 | Defines an IPv4 prefix list. |
| ipv6 prefix-list, on page 471 | Defines an IPv6 prefix list. |
| permit (prefix-list), on page 473 | Sets the permit conditions for an IPv4 or IPv6 prefix list. |
| remark (prefix-list), on page 476 | Inserts a helpful remark about a prefix list entry. |
| show prefix-list ipv4, on page 484 | Displays the contents of the current IPv4 prefix list. |
| show prefix-list ipv6, on page 487 | Displays the contents of the current IPv6 prefix list. |

# ipv4 prefix-list

To define an IP Version (IPv4) prefix list by name, use the **ipv4 prefix-list** command in XR Config mode. To remove the prefix list, use the **no** form of this command.

**ipv4 prefix-list** *name*
**no ipv4 prefix-list** *name*

## Syntax Description

| | |
|---|---|
| *name* | Name of the prefix list. Names cannot contain a space or quotation marks. |

## Command Default

No IPv4 prefix list is defined.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **ipv4 prefix-list** command to configure an IPv4 prefix list. This command places the router in prefix-list configuration mode, in which the denied or permitted access conditions must be defined with the **deny** or **permit** command. You must add a condition to create the prefix list.

Use the **resequence prefix-list ipv4** command to renumber existing statements and increment subsequent statements to allow a new IPv4 prefix list statement (**permit**, **deny**, or **remark**) to be added. Specify the first entry number (the *base*) and the increment by which to separate the entry numbers of the statements. The software will renumber the existing statements, thereby making room to add new statements with the unused entry numbers.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| ipv4 | read, write |

## Examples

The following example shows the prefix lists, then configures list2, then shows the conditions in both prefix lists:

```
RP/0/RP0/CPU0:router# show prefix-list ipv4

ipv4 prefix-list list1
 10 permit 172.20.10.171/16 le 24
 20 permit 172.18.0.0/16
 30 deny 172.24.20.164/16 ge 25

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list list2
```

```
RP/0/RP0/CPU0:router(config-ipv4_pfx)#deny 172.18.30.154/16 ge 25
RP/0/RP0/CPU0:router(config-ipv4_pfx)#
Uncommitted changes found, commit them? [yes]: Y

RP/0/RP0/CPU0:router# show prefix-list ipv4

ipv4 prefix-list list1
 10 permit 172.20.10.171/16 le 24
 20 permit 172.18.0.0/16
 30 deny 172.24.20.164/16 ge 25
ipv4 prefix-list list2
 10 deny 172.18.30.154/16 ge 25
```

## Related Commands

| Command | Description |
|---|---|
| deny (prefix-list), on page 466 | Sets deny conditions for an IPv4 or IPv6 prefix list. |
| permit (prefix-list), on page 473 | Sets permit conditions for an IPv4 or IPv6 prefix list. |
| remark (prefix-list), on page 476 | Inserts a helpful remark about a prefix list entry. |
| resequence prefix-list ipv4, on page 478 | Renumbers existing statements and increments subsequent statements. |
| show prefix-list ipv4, on page 484 | Displays the contents of the current IPv4 prefix list. |

# ipv6 prefix-list

To define an IP Version (IPv6) prefix list by name, use the **ipv6 prefix-list** command in XR Config mode. To remove the prefix list, use the **no** form of this command.

**ipv6 prefix-list** *name*
**no ipv6 prefix-list** *name*

## Syntax Description

| | |
|---|---|
| *name* | Name of the prefix list. Names cannot contain a space or quotation marks. |

## Command Default

No IPv6 prefix list is defined.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |
| ipv6 | read, write |

## Examples

The following example shows how to create a prefix list named list-1:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list list-1
RP/0/RP0/CPU0:router(config-ipv6-pfx)# 40 permit 2000:1::/64
RP/0/RP0/CPU0:router(config-ipv6-pfx)# 60 deny 3000:1::/64
RP/0/RP0/CPU0:router(config-ipv6-pfx)#
Uncommitted changes found, commit them? [yes]: y

RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list1
 40 permit 2000:1::/64
 60 deny 3000:1::/64
RP/0/RP0/CPU0:router#
```

## Related Commands

| Command | Description |
|---|---|
| deny (prefix-list), on page 466 | Sets deny conditions for an IPv4 or IPv6 prefix list. |

| Command | Description |
|---|---|
| permit (prefix-list), on page 473 | Sets permit conditions for an IPv4 or IPv6 prefix list. |
| remark (prefix-list), on page 476 | Inserts a helpful remark about a prefix list entry. |
| show prefix-list ipv6, on page 487 | Displays the contents of the current IPv6 prefix list. |

# permit (prefix-list)

To set permit conditions for an IP Version 4 (IPv4) or IP Version 6 (IPv6) prefix list, use the **permit** command in IPv4 prefix list configuration or IPv6 prefix list configuration modes. To remove a condition from a prefix list, use the **no** form of this command.

[*sequence-number*] **permit** *network/length* [**ge** *value*] [**le** *value*] [**eq** *value*]
**no** *sequence-number* **permit**

## Syntax Description

| | |
|---|---|
| *sequence-number* | (Optional) Number of the **permit** statement in the prefix list. This number determines the order of the statements in the prefix list. Range is 1 to 2147483646. By default, the first statement is number 10, and the subsequent statements are incremented by 10. |
| *network* / *length* | Network number and length (in bits) of the network mask. |
| **ge** *value* | (Optional) Specifies a prefix length greater than or equal to the value. It is the lowest value of a range of the *length* (the "from" portion of the length range). Range is 1 to 128. |
| **le** *value* | (Optional) Specifies a prefix length less than or equal to the value. It is the highest value of a range of the *length* (the "to" portion of the length range). Range is 1 to 128. |
| **eq** *value* | (Optional) Exact value of the *length* . Range is 1 to 128. |

## Command Default

No default behavior or value

## Command Modes

IPv4 prefix list configuration

IPv6 prefix list configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **permit** command to specify conditions under which a packet can pass the prefix list.

The **ge**, **le** and **eq** keywords can be used to specify the range of the prefix length to be matched, for prefixes that are more specific than the *network/length* argument. Exact match is assumed when neither **ge** nor **le** is specified. The range is assumed to be from the **ge** *value* to 32 if only the **ge** keyword is specified. The range is assumed to be from the *length* to the **le** *value* argument if only the **le** attribute is specified.

A specified **ge** *value* or **le** *value* must satisfy the following condition:

*length* < **ge** *value* < **le** *value* <= 32 (for IPv4)

*length* < **ge** *value* < **le** *value* <= 128 (for IPv6)

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

The following example shows how to permit the prefix 172.18.0.0/16:

```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list list1
RP/0/RP0/CPU0:router(config-ipv4_pfx)# permit 172.18.0.0/16
```

The following example shows how to accept a mask length of up to 24 bits in routes with the prefix 172.20.10.171/16:

```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list list1
RP/0/RP0/CPU0:router(config-ipv4_pfx)# permit 172.20.10.171/16 le 24
```

The following example shows how to permit mask lengths from 8 to 24 bits in all address space:

```
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list list1
RP/0/RP0/CPU0:router(config-ipv6_pfx)# permit 2000:1::/64 ge 8 le 24
```

The following example shows how to add permit conditions to list3, then remove the condition with the sequence number 30:

```
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list list3
RP/0/RP0/CPU0:router(config-ipv6_pfx)# permit 2000:1::/64 ge 25
RP/0/RP0/CPU0:router(config-ipv6_pfx)# permit 3000:1::/64 le 32
RP/0/RP0/CPU0:router(config-ipv6_pfx)# permit 3000:1::/64 ge 25
Uncommitted changes found, commit them? [yes]: y
RP/0/RP0/CPU0:router#show ipv6 prefix-list

ipv6 prefix-list list3
 10 permit 2000:1::/64 ge 25
 20 permit 3000:1::/64 le 32
 30 permit 4000:1::/64 ge 25

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list list3
RP/0/RP0/CPU0:router(config-ipv6_pfx)# no 30
Uncommitted changes found, commit them? [yes]: y
RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list3
 10 permit 2000:1::/64 ge 25
 20 permit 3000:1::/64 le 32

10 deny 2000:1::/64 ge 25
 20 deny 3000:1::/64 le 32
 30 deny 4000:1::/64 ge 25
```

## Related Commands

| Command | Description |
|---|---|
| deny (prefix-list), on page 466 | Sets deny conditions for an IPv4 or IPv6 prefix list. |
| ipv4 prefix-list, on page 469 | Creates an IPv4 prefix list. |
| ipv6 prefix-list, on page 471 | Creates an IPv6 prefix list. |
| remark (prefix-list), on page 476 | Inserts a helpful remark about a prefix list entry. |
| show prefix-list ipv4, on page 484 | Displays the contents of current IPv4 prefix lists. |
| show prefix-list ipv6, on page 487 | Displays the contents of current IPv6 prefix lists. |

# remark (prefix-list)

To write a helpful comment (remark) for an entry in either an IP Version 4 (IPv4) or IP Version 6 (IPv6) prefix list, use the **remark** command in IPv4 prefix-list configuration or IPv6 prefix-list configuration modes. To remove the remark, use the **no** form of this command.

[*sequence-number*] **remark** *remark*
**no** *sequence-number*

## Syntax Description

| | |
|---|---|
| *sequence-number* | (Optional) Number of the **remark** statement in the prefix list. This number determines the order of the statements in the prefix list. The number can be from 1 to 2147483646. (By default, the first statement is number 10, and the subsequent statements are incremented by 10). |
| *remark* | Comment that describes the entry in the prefix list, up to 255 characters long. |

## Command Default

The prefix list entries have no remarks.

## Command Modes

IPv4 prefix-list configuration

IPv6 prefix-list configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **remark** command to write a helpful comment for an entry in a prefix list. The remark can be up to 255 characters in length; anything longer is truncated.

If you know the sequence number of the remark you want to delete, you can remove it by entering the **no** *sequence-number* command.

Use the **resequence prefix-list ipv4** command if you want to add statements to an existing IPv4 prefix list.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

In the following example, a remark is made to explain a prefix list entry:

```
RP/0/RP0/CPU0:router(config)# ipv4 prefix-list deny-ten
RP/0/RP0/CPU0:router(config-ipv4_pfx)# 10 remark Deny all routes with a prefix of 10/8
RP/0/RP0/CPU0:router(config-ipv4_pfx)# 20 deny 10.0.0.0/8 le 32
RP/0/RP0/CPU0:router(config-ipv4_pfx)# end
```

In the following example, a remark is made to explain usage:

```
RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list1
 40 permit 2000:1::/64
 60 deny 3000:1::/64

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ipv6 prefix-list list1
RP/0/RP0/CPU0:router(config-ipv6-pfx)# 10 remark use from july23 forward
RP/0/RP0/CPU0:router(config-ipv6-pfx)#
Uncommitted changes found, commit them? [yes]: y

RP/0/0/CPU0:Apr  4 02:20:34.851 : config[65700]: %LIBTARCFG-6-COMMIT : Configura
tion committed by user 'UNKNOWN'.   Use 'show commit changes 1000000023' to view
 the changes.
RP/0/0/CPU0:Apr  4 02:20:34.984 : config[65700]: %SYS-5-CONFIG_I : Configured fr
om console by console
RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list1
 10 remark use from july23 forward
 40 permit 2000:1::/64
 60 deny 3000:1::/64
```

## Related Commands

| Command | Description |
|---|---|
| ipv4 prefix-list, on page 469 | Creates an entry in a prefix list. |
| resequence prefix-list ipv4, on page 478 | Renumbers existing statements and increments subsequent statements. |
| show prefix-list ipv4, on page 484 | Displays information about a prefix list or prefix list entries. |

# resequence prefix-list ipv4

To renumber existing statements and increment subsequent statements to allow a new prefix list statement (**permit**, **deny**, or **remark**) to be added, use the **resequence prefix-list ipv4** command in System Admin Config mode.

**resequence prefix-list ipv4** *name* [*base* [*increment*]]

## Syntax Description

| | |
|---|---|
| *name* | Name of a prefix list. |
| *base* | (Optional) Number of the first statement in the specified prefix list, which determines its order in the prefix list. Maximum value is 2147483646. |
| *increment* | (Optional) Number by which the base sequence number is incremented for subsequent statements. Maximum value is 2147483646. |

## Command Default

*base*: 10

*increment*: 10

## Command Modes

System Admin Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The sequence number of a prefix list entry determines the order of the entries in the list. The router compares network addresses to the prefix list entries. The router begins the comparison at the top of the prefix list, with the entry having the lowest sequence number.

If multiple entries of a prefix list match a prefix, the entry with the lowest sequence number is considered the real match. When a match or deny occurs, the router does not go through the rest of the prefix list.

By default, the first statement in a prefix list is sequence number 10, and the subsequent statements are incremented by 10.

Use the **resequence prefix-list ipv4** command to add a **permit**, **deny**, or **remark** statement between consecutive entries in an existing IPv4 prefix list. Specify the first entry number (the *base*) and the increment by which to separate the entry numbers of the statements. The software renumbers the existing statements, thereby making room to add new statements with the unused entry numbers.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

The following example shows how to display the sequence number intervals for prefix list list1, resequence list1 from 10 to 30, and displays the resulting sequence numbers:

```
RP/0/RP0/CPU0:router# show prefix-list ipv4

ipv4 prefix-list list1
 10 permit 172.20.10.171/16 le 24
 20 permit 172.18.0.0/16
 30 deny 172.24.20.164/16 ge 25
ipv4 prefix-list list2
 10 deny 172.18.30.154/16 ge 25

RP/0/RP0/CPU0:router# resequence prefix-list ipv4 list1 10 30

RP/0/0/CPU0:Apr  4 02:29:39.513 : ipv4_acl_action_edm[183]: %LIBTARCFG-6-COMMIT
: Configuration committed by user 'UNKNOWN'.   Use 'show commit changes 10000000
24' to view the changes.

RP/0/RP0/CPU0:router# show prefix-list ipv4

ipv4 prefix-list list1
 10 permit 172.20.10.171/16 le 24
 40 permit 172.18.0.0/16
 70 deny 172.24.20.164/16 ge 25
ipv4 prefix-list list2
 10 deny 172.18.30.154/16 ge 25
```

## Related Commands

| Command | Description |
|---|---|
| deny (prefix-list), on page 466 | Sets deny conditions for an IPv4 or IPv6 prefix list. |
| permit (prefix-list), on page 473 | Sets permit conditions for an IPv4 or IPv6 prefix list. |
| remark (prefix-list), on page 476 | Inserts a helpful remark about a prefix list entry. |
| show prefix-list ipv4, on page 484 | Displays the contents of the current IPv4 prefix list. |

# resequence prefix-list ipv6

To renumber existing statements and increment subsequent statements to allow a new prefix list statement (**permit**, **deny**, or **remark**) to be added, use the **resequence prefix-list ipv6** command in XR EXEC mode.

**resequence prefix-list ipv6** *name* [*base* [*increment*]]

## Syntax Description

| | |
|---|---|
| *name* | Name of a prefix list. |
| *base* | (Optional) Number of the first statement in the specified prefix list, which determines its order in the prefix list. Maximum value is 2147483644. |
| *increment* | (Optional) Number by which the base sequence number is incremented for subsequent statements. Maximum value is 2147483644. |

## Command Default

*base*: 10

*increment*: 10

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The sequence number of a prefix list entry determines the order of the entries in the list. The router compares network addresses to the prefix list entries. The router begins the comparison at the top of the prefix list, with the entry having the lowest sequence number.

If multiple entries of a prefix list match a prefix, the entry with the lowest sequence number is considered the real match. Once a match or deny occurs, the router does not go through the rest of the prefix list.

By default, the first statement in a prefix list is sequence number 10, and the subsequent statements are incremented by 10.

Use the **resequence prefix-list ipv6** command to add a **permit**, **deny**, or **remark** statement between consecutive entries in an existing IPv6 prefix list. Specify the first entry number (the *base*) and the increment by which to separate the entry numbers of the statements. The software renumbers the existing statements, thereby making room to add new statements with the unused entry numbers.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read, write |

## Examples

The following example shows how to display the sequence number intervals for prefix list 1, resequence list1 from 10 to 30, and displays the resulting sequence numbers:

```
RP/0/RP0/CPU0:router# show prefix-list ipv6

prefix-list list1
 10 permit
/16 le 24
 20 permit 3000:1::/16 le 32
 20 permit 172.18.0.0/16
 30 deny
/16 ge 25

prefix-list list2
 10 deny
/16 ge 25

RP/0/RP0/CPU0:router# resequence prefix-list ipv4 list1 10 30

RP/0//CPU0:
Apr  4 02:29:39.513 :
[183]: %LIBTARCFG-6-COMMIT
: Configuration committed by user 'UNKNOWN'.   Use 'show commit changes 10000000
24' to view the changes.
```

# show prefix-list

To display information about a prefix list or prefix list entries, use the **show prefix-list** command in XR EXEC mode.

**show prefix-list** [*list-name*] [*sequence-number*]

**Syntax Description**

| | |
|---|---|
| *list-name* | (Optional) Name of a prefix list. |
| *sequence-number* | (Optional) Sequence number of the prefix list entry. Range is 1 to 2147483646. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operations |
|---|---|
| acl | read |

**Examples**

The following sample output is from the **show prefix-list** command:

```
RP/0/RP0/CPU0:router# show prefix-list
```

# show prefix-list afi-all

To display the contents of the prefix list for all the address families, use the **show prefix-list afi-all** command in XR EXEC mode.

**show prefix-list afi-all**

## Syntax Description

This command has no keywords or arguments.

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

The following sample output is from the **show prefix-list afi-all** command:

```
RP/0/RP0/CPU0:router# show prefix-list afi-all
```

# show prefix-list ipv4

To display the contents of current IP Version 4 (IPv4) prefix list, use the **show prefix-list ipv4** command in XR EXEC mode.

**show prefix-list ipv4** [**summary**] [*list-name*] [*sequence-number*]

## Syntax Description

| | |
|---|---|
| *list-name* | (Optional) Name of a prefix list. |
| *sequence-number* | (Optional) Sequence number of the prefix list entry. Range is 1 to 2147483646. |
| **summary** | (Optional) Displays summary output of prefix list contents. |

## Command Default

All IPv4 prefix lists are displayed.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **show prefix-list ipv4** command to display the contents of all IPv4 prefix lists. To display the contents of a specific IPv4 prefix list, use the *name* argument. Use the *sequence-number* argument to specify a given prefix list entry. Use the **summary** keyword to display a summary of prefix list contents.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

The following example displays all configured prefix lists:

```
RP/0/RP0/CPU0:router# show prefix-list ipv4

ipv4 prefix-list list1
 10 permit 172.20.10.171/16 le 24
 20 permit 172.18.0.0/16
 30 deny 172.24.20.164/16 ge 25
ipv4 prefix-list list2
 10 deny 172.18.30.154/16 ge 25
```

The following example uses the *list-name* argument to display the prefix list named list1:

```
RP/0/RP0/CPU0:router# show prefix-list ipv4 list1

ipv4 prefix-list list1
 10 permit 172.20.10.171/16 le 24
 20 permit 172.18.0.0/16
```

```
 30 deny 172.24.20.164/16 ge 25
```

The following example uses the *list-name* and *sequence-number* argument to display a prefix list named list1 with a sequence number of 10:

```
RP/0/RP0/CPU0:router# show prefix-list ipv4 list1 30

ipv4 prefix-list list1
 30 deny 172.24.20.164/16 ge 25
```

## Related Commands

| Command | Description |
|---|---|
| clear prefix-list ipv4, on page 458 | Resest the hit count on an IPv4 prefix list. |
| ipv4 prefix-list, on page 469 | Defines an IPv4 prefix list. |
| show prefix-list ipv6, on page 487 | Displays the contents of the current IPv6 prefix list. |

# show prefix-list ipv4 standby

To display the contents of current IPv4 standby access lists, use the **show access-lists ipv4 standby** command in XR EXEC mode.

**show prefix-list ipv4 standby** [**summary**] [*prefix-list name*]

## Syntax Description

| | |
|---|---|
| *prefix-list name* | (Optional) Name of a particular IPv4 prefix list. The value of the prefix-list-name argument is a string of alphanumeric characters that cannot include spaces or quotation marks. |
| **summary** | (Optional) Displays a summary of all current IPv4 standby prefix lists. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **show prefix-list ipv4 standby** command to display the contents of current IPv4 standby prefix lists. To display the contents of a specific IPv4 prefix list, use the *name* argument.

Use the **show prefix-list ipv4 standby summary** command to display a summary of all standby IPv4 prefix lists.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

In the following example, the contents of all IPv4 access lists are displayed:

```
RP/0/RP0/CPU0:router# show prefix-list ipv4 standby summary
Prefix List Summary:
  Total Prefix Lists configured:         2
  Total Prefix List entries configured : 6
```

# show prefix-list ipv6

To display the contents of the current IP Version 6 (IPv6) prefix list, use the **show prefix-list ipv6** command in XR EXEC mode.

**show prefix-list ipv6** [**summary**] [*list-name*] [*sequence-number*]

## Syntax Description

| | |
|---|---|
| *list-name* | (Optional) Name of a prefix list. |
| *sequence-number* | (Optional) Sequence number of the prefix list entry. Range is 1 to 2147483646. |
| **summary** | (Optional) Displays summary output of prefix list contents. |

## Command Default

All IPv6 prefix lists are displayed.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **show prefix-list ipv6** command to display the contents of all IPv4 prefix lists.

To display the contents of a specific IPv6 prefix list, use the *name* argument. Use the *sequence-number* argument to specify a given prefix list entry. Use the **summary** keyword to display a summary of prefix list contents.

## Task ID

| Task ID | Operations |
|---|---|
| acl | read |

## Examples

The following example shows how to display all configured prefix lists:

```
RP/0/RP0/CPU0:router# show prefix-list ipv6

ipv6 prefix-list list1
 10 permit 5555::/24
 20 deny 3000::/24
 30 permit 2000::/24
ipv6 prefix-list list2
 10 permit 2000::/24
```

The following example uses the *list-name* argument to display the prefix list named list1:

```
RP/0/RP0/CPU0:router# show prefix-list ipv6 list1

ipv6 prefix-list list1
 10 permit 5555::/24
 20 deny 3000::/24
```

```
 30 permit 2000::/24
```

The following example uses the *list-name* and *sequence-number* argument to display a prefix list named list1 with a sequence number of 10:

```
RP/0/RP0/CPU0:router# show prefix-list ipv6 list1 10

ipv6 prefix-list abc
 10 permit 5555::/24
```

The following example displays a summary of prefix list contents:

```
RP/0/RP0/CPU0:router# show prefix-list ipv6 summary

Prefix List Summary:
  Total Prefix Lists configured:        2
  Total Prefix List entries configured: 2
```

## Related Commands

| Command | Description |
|---|---|
| clear prefix-list ipv6 , on page 460 | Resest the hit count on an IPv4 prefix list. |
| copy prefix-list ipv6 , on page 464 | Creates a copy of an existing IPv6 prefix list. |
| ipv6 prefix-list, on page 471 | Creates an IPv6 prefix list. |

# Transport Stack Commands

This chapter describes the Cisco IOS XR softwarecommands used to configure and monitor features related to the transport stack (Nonstop Routing [NSR ], TCP, User Datagram Protocol [UDP], and RAW ). Any IP protocol other than TCP or UDP is known as a *RAW* protocol.

For detailed information about transport stack concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# clear nsr ncd client

To clear the counters of a specified client or all the clients of nonstop routing (NSR) Consumer Demuxer (NCD), use the **clear nsr ncd client** command in XR EXEC mode.

**clear nsr ncd client** {*PID value* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *PID value* | Process ID value of the client in which counters need to be cleared. The range is from 0 to 4294967295. |
| **all** | Clears the counters for all NCD clients. |
| **location** *node-id* | (Optional) Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

The default value for the *node-id* argument is the current node in which the command is being executed. The *PID value* argument does not have a default value.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

The active and standby instances of some NSR-capable applications communicate through two queues, and these applications are multiplexed onto these queues. NSR consumer demuxer (NCD) is a process that provides the demuxing services on the receiver side.

You can use the **clear nsr ncd client** command to troubleshoot traffic issues. If you clear the existing counters, it can help you to monitor the delta changes.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows how to clear all the counters for all NCD clients:

```
RP/0/RP0/CPU0:router# clear nsr ncd client all
RP/0/RP0/CPU0:router# show nsr ncd client all

Client PID                          : 3874979
Client Protocol                     : TCP
Client Instance                     : 1
Total packets received              : 0
Total acks received                 : 0
Total packets/acks accepted         : 0
Errors in changing packet ownership  : 0
Errors in setting application offset : 0
```

```
Errors in enqueuing to client        : 0
Time of last clear                   : Sun Jun 10 14:43:44 20

RP/0/RP0/CPU0:router# show nsr ncd client brief

                              Total    Total   Accepted
Pid    Protocol   Instance    Packets Acks    Packets/Acks
3874979   TCP          1          0   0         0
```

## Related Commands

| Command | Description |
|---|---|
| clear nsr ncd queue, on page 493 | Clears the counters for the NSR Consumer Demuxer (NCD) queue. |
| show nsr ncd client, on page 521 | Displays information about the clients for NSR Consumer Demuxer (NCD). |
| show nsr ncd queue, on page 523 | Displays information about the nonstop routing (NSR) Consumer Queue and Dispatch (QAD) queues. |

# clear nsr ncd queue

To clear the counters for the nonstop routing (NSR) Consumer Demuxer (NCD) queue, use the **clear nsr ncd queue** command in XR EXEC mode.

**clear nsr ncd queue** {**all** | **high** | **low**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **all** | Clears the counters for all the NCD queues. |
| **high** | Clears the counters for the high-priority NCD queue. |
| **low** | Clears the counters the low-priority NCD queue. |
| **location** *node-id* | (Optional) Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows how to clear the counters for all the NCD queues:

```
RP/0/RP0/CPU0:router# clear nsr ncd queue all
RP/0/RP0/CPU0:router# show nsr ncd queue all

Queue Name                          : NSR_LOW
Total packets received              : 0
Total packets accepted              : 0
Errors in getting datagram offset   : 0
Errors in getting packet length     : 0
Errors in calculating checksum      : 0
Errors due to bad checksum          : 0
Errors in reading packet data       : 0
Errors due to bad NCD header        : 0
Drops due to a non-existent client  : 0
Errors in changing packet ownership : 0
Errors in setting application offset : 0
Errors in enqueuing to client       : 0
Time of last clear                  : Sun Jun 10 14:44:38 2007
```

```
Queue Name                          : NSR_HIGH
Total packets received              : 0
Total packets accepted              : 0
Errors in getting datagram offset   : 0
Errors in getting packet length     : 0
Errors in calculating checksum      : 0
Errors due to bad checksum          : 0
Errors in reading packet data       : 0
Errors due to bad NCD header        : 0
Drops due to a non-existent client  : 0
Errors in changing packet ownership : 0
Errors in setting application offset : 0
Errors in enqueuing to client       : 0
Time of last clear                  : Sun Jun 10 14:44:38 2007

RP/0/RP0/CPU0:router# show nsr ncd queue brief

                          Total        Accepted
          Queue         Packets         Packets
        NSR_LOW               0               0
       NSR_HIGH               0               0
```

## Related Commands

| Command | Description |
|---|---|
| clear nsr ncd client, on page 491 | Clears the counters for the NSR Consumer Demuxer (NCD) client. |
| nsr process-failures switchover, on page 516 | Configures failover as a recovery action for active instances to switch over to a standby route processor (RP)to maintain nonstop routing (NSR). |
| show nsr ncd client, on page 521 | Displays information about the clients for NSR Consumer Demuxer (NCD). |
| show nsr ncd queue, on page 523 | Displays information about the nonstop routing (NSR) Consumer Queue and Dispatch (QAD) queues. |

# clear raw statistics pcb

To clear statistics for a single RAW connection or for all RAW connections, use the **clear raw statistics pcb** command in XR EXEC mode.

**clear raw statistics pcb** {**all** | **pcb-address**} [**location node-id**]

## Syntax Description

| | |
|---|---|
| all | Clears statistics for all RAW connections. |
| pcb-address | Clears statistics for a specific RAW connection. |
| **location** *node-id* | Clears statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **all** keyword to clear all RAW connections. To clear a specific RAW connection, enter the protocol control block (PCB) address of the RAW connection. **Use the show raw brief** command to obtain the PCB address.

Use the **location** keyword and *node-id* argument to clear RAW statistics for a designated node.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows how to clear statistics for a RAW connection with PCB address 0x80553b0:

```
RP/0/RP0/CPU0:router# clear raw statistics pcb 0x80553b0
RP/0/RP0/CPU0:router# show raw statistics pcb 0x80553b0

Statistics for PCB 0x80553b0
Send: 0 packets received from application
0 xipc pulse received from application
0 packets sent to network
0 packets failed getting queued to network
Rcvd: 0 packets received from network
0 packets queued to application
0 packets failed queued to application
```

The following example shows how to clear statistics for all RAW connections:

```
RP/0/RP0/CPU0:router# clear raw statistics pcb all
RP/0/RP0/CPU0:router# show raw statistics pcb all

Statistics for PCB 0x805484c
Send: 0 packets received from application
0 xipc pulse received from application
0 packets sent to network
0 packets failed getting queued to network
Rcvd: 0 packets received from network
0 packets queued to application
0 packets failed queued to application

Statistics for PCB 0x8054f80
Send: 0 packets received from application
0 xipc pulse received from application
0 packets sent to network
0 packets failed getting queued to network
Rcvd: 0 packets received from network
0 packets queued to application
0 packets failed queued to application

Statistics for PCB 0x80553b0
Send: 0 packets received from application
0 xipc pulse received from application
0 packets sent to network
0 packets failed getting queued to network
Rcvd: 0 packets received from network
0 packets queued to application
0 packets failed queued to application
```

## Related Commands

| Command | Description |
|---|---|
| show raw brief, on page 525 | Displays information about active RAW IP sockets. |
| show raw statistics pcb, on page 531 | Displays statistics for either a single RAW connection or all RAW connections. |

# clear tcp nsr client

To bring the nonstop routing (NSR) down on all the sessions that are owned by the specified client, use the **clear tcp nsr client** command in XR EXEC mode.

**clear tcp nsr client** {*ccb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *ccb-address* | Client Control Block (CCB) of the NSR client. |
| **all** | Specifies all the clients. |
| **location** *node-id* | (Optional) Displays client information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

The location defaults to the current node in which the command is executing.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

The output of the **show tcp nsr client** command is used to locate the CCB of the desired client.

Use the **clear tcp nsr client** command to gracefully bring down NSR session that are owned by one client or all clients. In addition, the **clear tcp nsr client** command is used as a work around if the activity on the sessions freezes.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows that the nonstop routing (NSR) client is cleared for 0x482afacc The two sessions had NSR already up before executing the **clear tcp nsr client** command. NSR is no longer up after executing the **clear tcp nsr client** command.

```
RP/0/RP0/CPU0:router# show tcp nsr client brief

CCB          Proc Name    Instance     Sets       Sessions/NSR Up Sessions
0x482c10e0   mpls_ldp          1            2          3/1
0x482afacc   mpls_ldp          2            1          2/2

RP/0/RP0/CPU0:router# clear tcp nsr client 0x482afacc
RP/0/RP0/CPU0:router# show tcp nsr client brief

CCB          Proc Name    Instance     Sets       Sessions/NSR Up Sessions
0x482c10e0   mpls_ldp          1            2          3/1
0x482afacc   mpls_ldp          2            1          2/0
```

## Related Commands

| Command | Description |
|---|---|
| nsr process-failures switchover, on page 516 | Configures failover as a recovery action for active instances to switch over to a standby route processor (RP) to maintain nonstop routing (NSR). |
| show tcp nsr client brief, on page 542 | Displays brief information about the state of nonstop routing (NSR) of TCP clients on different nodes. |

# clear tcp nsr pcb

To bring the nonstop routing (NSR) down on a specified connection or all connections, use the **clear tcp nsr pcb** command in XR EXEC mode.

**clear tcp nsr pcb** {*pcb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| pcb-address | PCB address range for the specific connection information. 0 to ffffffff. For example, the address range can be 0x482a4e20. |
| all | Specifies all the connections. |
| **location** *node-id* | (Optional) Displays connection information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## 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 **location** keyword is used so that active and standby TCP instances are independently queried.

The output of the **show tcp nsr brief** command is used to locate the Protocol Control Block (PCB) of a desired connection.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows that the information for TCP connections is cleared:

```
RP/0/RP0/CPU0:router# show tcp nsr brief

PCB     Local Address Foreign Address        NSR   RcvOnly
0x482d7470
.1.1.1:646
.1.1.2:14142          Up    No
0x482d2844
.1.1.1:646
.1.1.2:15539          Up    No
0x482d3bc0
.1.1.1:646
.1.1.2:25671          Up    No
0x482d4f3c
```

```
.1.1.1:646
.1.1.2:32319          Up    No
0x482d87ec
.1.1.1:646
.1.1.2:39592          Up    No
0x482cd670
.1.1.1:646
.1.1.2:43447          Up    No
0x482d14c8
.1.1.1:646
.1.1.2:45803          Up    No
0x482bdee4
.1.1.1:646
.1.1.2:55844          Up    No
0x482d62b8
.1.1.1:646
.1.1.2:60695          Up    No
0x482d0310
.1.1.1:646
.1.1.2:63007          Up    No

RP/0/RP0/CPU0:router# clear tcp nsr pcb 0x482d7470
RP/0/RP0/CPU0:router# clear tcp nsr pcb 0x482d2844
RP/0/RP0/CPU0:router# show tcp nsr brief

PCB     Local Address Foreign Address NSR   RcvOnly
0x482d7470
.1.1.1:646
.1.1.2:14142          Down  No
0x482d2844
.1.1.1:646
.1.1.2:15539          Down  No
0x482d3bc0
.1.1.1:646
.1.1.2:25671          Up    No
0x482d4f3c
.1.1.1:646
.1.1.2:32319          Up    No
0x482d87ec
.1.1.1:646
.1.1.2:39592          Up    No
0x482cd670
.1.1.1:646
.1.1.2:43447          Up    No
0x482d14c8
.1.1.1:646
.1.1.2:45803          Up    No
0x482bdee4
.1.1.1:646
.1.1.2:55844          Up    No
0x482d62b8
.1.1.1:646
.1.1.2:60695          Up    No
0x482d0310
.1.1.1:646
.1.1.2:63007          Up    No
```

## Related Commands

| Command | Description |
|---|---|
| show tcp nsr brief, on page 540 | Displays the key nonstop routing (NSR) state of TCP connections on different nodes. |

| Command | Description |
| --- | --- |
| show tcp nsr detail pcb, on page 546 | Displays detailed information about the state of nonstop routing (NSR) for TCP connections. |

# clear tcp nsr session-set

To clear the nonstop routing (NSR) on all the sessions in the specified session-set or all session sets, use the **clear tcp nsr session-set** command in XR EXEC mode.

**clear tcp nsr session-set** { *sscb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *sscb-address* | Session-Set Control Block (SSCB) address range for the specific session set information. 0 to ffffffff. For example, the address range can be 0x482a4e20. |
| **all** | Specifies all the session sets. |
| **location** *node-id* | (Optional) Displays session set information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

The output of the **show tcp nsr session-set brief** command is used to locate the SSCB of the desired session-set.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows that the information for the session sets is cleared:

```
RP/0/RP0/CPU0:router# show tcp nsr client brief

CCB               Proc Name         Instance  Sets       Sessions/NSR Up Sessions
0x482b5ee0        mpls_ldp             1        1          10/10

RP/0/RP0/CPU0:router# clear tcp nsr client 0x482b5ee0
RP/0/RP0/CPU0:router# show tcp nsr client brief

CCB               Proc Name         Instance  Sets       Sessions/NSR Up Sessions
0x482b5ee0        mpls_ldp             1        1          10/0
```

## Related Commands

| Command | Description |
|---|---|
| show tcp nsr detail session-set, on page 549 | Displays detailed information about the nonstop routing (NSR) state of the session sets on different nodes. |

| Command | Description |
| --- | --- |
| show tcp nsr session-set brief, on page 551 | Displays brief information about the session sets for the state of nonstop routing (NSR) on different nodes. |

# clear tcp nsr statistics client

To clear the nonstop routing (NSR) statistics of the client, use the **clear tcp nsr statistics client** command in XR EXEC mode.

**clear tcp nsr statistics client** {*ccb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *ccb-address* | Client Control Block (CCB) of the desired client. For example, the address range can be 0x482a4e20. |
| **all** | Specifies all the clients. |
| **location** *node-id* | (Optional) Displays client information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## 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 **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows that the statistics for the NSR clients is cleared:

```
RP/0/RP0/CPU0:router# show tcp nsr statistics client all

============================================================
CCB: 0x482b5ee0
Name: mpls_ldp, Job ID: 365
Connected at: Thu Aug 16 18:20:32 2007

Notification Statistics :   Queued    Failed    Delivered Dropped
Init-Sync Done          :         2          0          2          0
Replicated Session Ready:         0          0          0          0
Operational Down        :        12          0         12          0
Last clear at: Never Cleared

RP/0/RP0/CPU0:router# clear tcp nsr statistics client all

RP/0/RP0/CPU0:router# show tcp nsr statistics client all
```

```
============================================================
CCB: 0x482b5ee0
Name: mpls_ldp, Job ID: 365
Connected at: Thu Aug 16 18:20:32 2007

Notification Statistics :   Queued    Failed    Delivered Dropped
Init-Sync Done          :         0         0          0         0
Replicated Session Ready:         0         0          0         0
Operational Down        :         0         0          0         0
Last clear at: Thu Aug 16 18:28:38 2007
```

## Related Commands

| Command | Description |
|---|---|
| show tcp nsr statistics client, on page 553 | Displays the nonstop routing (NSR) statistics for the client. |

# clear tcp nsr statistics pcb

To clear the nonstop routing (NSR) statistics for TCP connections, use the **clear tcp nsr statistics pcb** command in XR EXEC mode.

**clear tcp nsr statistics pcb** {*pcb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *pcb-address* | PCB address range for the specific connection information. 0 to ffffffff. For example, the address range can be 0x482a4e20. |
| **all** | Specifies all the connections. |
| **location** *node-id* | (Optional) Displays connection information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows that the NSR statistics for TCP connections is cleared:

```
RP/0/RP0/CPU0:router# show tcp nsr statistics pcb 0x482d14c8

==============================================================
PCB 0x482d14c8
Number of times NSR went up: 1
Number of times NSR went down: 0
Number of times NSR was disabled: 0
Number of times switch-over occured : 0
IACK RX Message Statistics:
        Number of iACKs dropped because SSO is not up              : 0
        Number of stale iACKs dropped                              : 1070
        Number of iACKs not held because of an immediate match     : 98
TX Messsage Statistics:
        Data transfer messages:
            Sent 317, Dropped 0, Data (Total/Avg.) 2282700/7200
            Rcvd 0
                Success           : 0
                Dropped (Trim)    : 0
        Segmentation instructions:
            Sent 1163, Dropped 0, Units (Total/Avg.) 4978/4
```

```
            Rcvd 0
                Success           : 0
                Dropped (Trim)    : 0
                Dropped (TCP)     : 0
        NACK messages:
            Sent 0, Dropped 0
            Rcvd 0
                Success           : 0
                Dropped (Data snd): 0
        Cleanup instructions      :
            Sent 8, Dropped 0
            Rcvd 0
                Success           : 0
                Dropped (Trim)    : 0
Last clear at: Never cleared

RP/0/RP0/CPU0:router# clear tcp nsr statistics pcb 0x482d14c8
RP/0/RP0/CPU0:router# show tcp nsr statistics pcb 0x482d14c8

==============================================================
PCB 0x482d14c8
Number of times NSR went up: 0
Number of times NSR went down: 0
Number of times NSR was disabled: 0
Number of times switch-over occured : 0
IACK RX Message Statistics:
        Number of iACKs dropped because SSO is not up               : 0
        Number of stale iACKs dropped                               : 0
        Number of iACKs not held because of an immediate match      : 0
TX Messsage Statistics:
        Data transfer messages:
            Sent 0, Dropped 0, Data (Total/Avg.) 0/0
            Rcvd 0
                Success           : 0
                Dropped (Trim)    : 0
        Segmentation instructions:
            Sent 0, Dropped 0, Units (Total/Avg.) 0/0
            Rcvd 0
                Success           : 0
                Dropped (Trim)    : 0
                Dropped (TCP)     : 0
        NACK messages:
            Sent 0, Dropped 0
            Rcvd 0
                Success           : 0
                Dropped (Data snd): 0
        Cleanup instructions      :
            Sent 0, Dropped 0
            Rcvd 0
                Success           : 0
                Dropped (Trim)    : 0
Last clear at: Thu Aug 16 18:32:12 2007
```

## Related Commands

| Command | Description |
|---|---|
| show tcp nsr statistics pcb, on page 555 | Displays the nonstop routing (NSR) statistics for a given Protocol Control Block (PCB). |

# clear tcp nsr statistics session-set

To clear the nonstop routing (NSR) statistics for session sets, use the **clear tcp nsr statistics session-set** command in XR EXEC mode.

**clear tcp nsr statistics session-set** {*sscb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *sscb-address* | Session-Set Control Block (SSCB) address range for the specific session set information. 0 to ffffffff. For example, the address range can be 0x482a4e20. |
| **all** | Specifies all the session sets. |
| **location** *node-id* | (Optional) Displays session set information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## 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 **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows that the NSR statistics for session sets is cleared:

```
RP/0/RP0/CPU0:router# show tcp nsr statistics session-set all

==================Session Set Stats ==========================
SSCB 0x482b6684, Set ID: 1
Number of times init-sync was attempted  :3
Number of times init-sync was successful :3
Number of times init-sync failed         :0
Number of times switch-over occured      :0
Last clear at: Never Cleared

RP/0/RP0/CPU0:router# clear tcp nsr statistics session-set all
RP/0/RP0/CPU0:router# show tcp nsr statistics session-set all

==================Session Set Stats ==========================
SSCB 0x482b6684, Set ID: 1
Number of times init-sync was attempted  :0
```

```
Number of times init-sync was successful :0
Number of times init-sync failed         :0
Number of times switch-over occured      :0
Last clear at: Thu Aug 16 18:37:00 2007
```

## Related Commands

| Command | Description |
|---|---|
| show tcp nsr statistics session-set, on page 557 | Displays nonstop routing (NSR) statistics for a session set. |

# clear tcp nsr statistics summary

To clear the nonstop routing (NSR) statistics summary, use the **clear tcp nsr statistics summary** command in XR EXEC mode.

**clear tcp nsr statistics summary** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays statistics summary information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows how to clear the summary statistics:

```
RP/0/RP0/CPU0:router# clear tcp nsr statistics summary
```

## Related Commands

| Command | Description |
|---|---|
| show tcp nsr statistics summary, on page 559 | Displays the nonstop routing (NSR) summary statistics across all TCP sessions. |

# clear tcp pcb

To clear TCP protocol control block (PCB) connections, use the **clear tcp pcb** command in XR EXEC mode.

**clear tcp pcb** {*pcb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *pcb-address* | Clears the TCP connection at the specified PCB address. |
| **all** | Clears all open TCP connections. |
| **location** *node-id* | (Optional) Clears the TCP connection for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **clear tcp pcb** command is useful for clearing hung TCP connections. Use the show tcp brief, on page 533 command to find the PCB address of the connection you want to clear.

If the **clear tcp pcb all** command is used, the software does not clear a TCP connection that is in the listen state. If a specific PCB address is specified, then a connection in listen state is cleared.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows that the TCP connection at PCB address 60B75E48 is cleared:

```
RP/0/RP0/CPU0:router# clear tcp pcb 60B75E48
```

## Related Commands

| Command | Description |
|---|---|
| show tcp brief, on page 533 | Displays the TCP summary table. |

# clear tcp statistics

To clear TCP statistics, use the **clear tcp statistics** command in XR EXEC mode.

**clear tcp statistics** {**pcb** {**all** *pcb-address*} | **summary**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **pcb all** | (Optional) Clears statistics for all TCP connections. |
| **pcb** *pcb-address* | (Optional) Clears statistics for a specific TCP connection. |
| **summary** | (Optional) Clears summary statistic for a specific node or connection. |
| **location** *node-id* | (Optional) Clears TCP statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **clear tcp statistics** command to clear TCP statistics. Use the show tcp statistics, on page 538 command to display TCP statistics. You might display TCP statistics and then clear them before you start debugging TCP.

The optional **location** keyword and *node-id* argument can be used to clear TCP statistics for a designated node.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows how to clear TCP statistics:

```
RP/0/RP0/CPU0:router# clear tcp statistics
```

## Related Commands

| Command | Description |
|---|---|
| show tcp statistics, on page 538 | Displays TCP statistics. |

# clear udp statistics

To clear User Datagram Protocol (UDP) statistics, use the **clear udp statistics** command in XR EXEC mode.

**clear udp statistics** {**pcb** {**all** *pcb-address*} | **summary**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| pcb all | Clears statistics for all UDP connections. |
| **pcb** *pcb-address* | Clears statistics for a specific UDP connection. |
| summary | Clears UDP summary statistics. |
| **location** *node-id* | Clears UDP statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **clear udp statistics** command to clear UDP statistics. Use the show udp statistics, on page 566 command to display UDP statistics. You might display UDP statistics and then clear them before you start debugging UDP.

The optional **location** keyword and *node-id* argument can be used to clear UDP statistics for a designated node.

## Task ID

| Task ID | Operations |
|---|---|
| transport | execute |

## Examples

The following example shows how to clear UDP summary statistics:

```
RP/0/RP0/CPU0:router# clear udp statistics summary
```

## Related Commands

| Command | Description |
|---|---|
| show udp statistics, on page 566 | Displays UDP statistics. |

# forward-protocol udp

To configure the system to forward any User Datagram Protocol (UDP) datagrams that are received as broadcast packets to a specified helper address, use the **forward-protocol udp** command in XR Config mode. To restore the system to its default condition with respect to this command, use the **no** form of this command.

**forward-protocol udp** {*port-number* | **disable** | **domain** | **nameserver** | **netbios-dgm** | **netbios-ns** | **tacacs** | **tftp**}
**no forward-protocol udp** {*port-number* | **disable** | **domain** | **nameserver** | **netbios-dgm** | **netbios-ns** | **tacacs** | **tftp**}

## Syntax Description

| | |
|---|---|
| *port-number* | Forwards UDP broadcast packets to a specified port number. Range is 1 to 65535. |
| **disable** | Disables IP Forward Protocol UDP. |
| **domain** | Forwards UDP broadcast packets to Domain Name Service (DNS, 53). |
| **nameserver** | Forwards UDP broadcast packets to IEN116 name service (obsolete, 42). |
| **netbios-dgm** | Forwards UDP broadcast packets to NetBIOS datagram service (138). |
| **netbios-ns** | Forwards UDP broadcast packets to NetBIOS name service (137). |
| **tacacs** | Forwards UDP broadcast packets to TACACS (49). |
| **tftp** | Forwards UDP broadcast packets to TFTP (69). |

## Command Default

Enabled

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **forward-protocol udp** command to specify that UDP broadcast packets received on the incoming interface are forwarded to a specified helper address.

When you configure the **forward-protocol udp** command, you must also configure the **helper-address** command to specify a helper address on an interface. The helper address is the IP address to which the UDP datagram is forwarded. Configure the **helper-address** command with IP addresses of hosts or networking devices that can handle the service. Because the helper address is configured per interface, you must configure a helper address for each incoming interface that will be receiving broadcasts that you want to forward.

You must configure one **forward-protocol udp** command per UDP port you want to forward. The port on the packet is either port 53 (**domain**), port 69 (**tftp**), or a port number you specify.

The **forward-protocol udp** command is by default enabled on the following ports: domain, nameserver, netbios-dgm, netbios-ns, tacacs, tftp. This feature can be disabled using the **forward-protocol udp disable** command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read, write |

## Examples

The following example shows how to specify that all UDP broadcast packets with port 53 or port 69 received on incoming MgmtEth interface 0/0/CPU0/0 are forwarded to 172.16.0.1. MgmtEth interface 0/0/CPU0/0 receiving the UDP broadcasts is configured with a helper address of 172.16.0.1, the destination address to which the UDP datagrams are forwarded.

```
RP/0/RP0/CPU0:router(config)# forward-protocol udp domain disable
RP/0/RP0/CPU0:router(config)# forward-protocol udp tftp disable
RP/0/RP0/CPU0:router(config)# interface MgmtEth 0/0/CPU0/0
RP/0/RP0/CPU0:router(config-if)# ipv4 helper-address 172.16.0.1
```

# nsr process-failures switchover

To configure failover as a recovery action for active instances to switch over to a standby route processor (RP) to maintain nonstop routing (NSR), use the **nsr process-failures switchover** command in XR Config mode. To disable this feature, use the **no** form of this command.

**nsr process-failures switchover**
**no nsr process-failures switchover**

## Syntax Description

This command has no keywords or arguments.

## Command Default

If not configured, a process failure of the active TCP or its applications (for example LDP, BGP, and so forth) can cause sessions to go down, and NSR is not provided.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read, write |

## Examples

The following example shows how to use the **nsr process-failures switchover** command:

```
RP/0/RP0/CPU0:router(config)# nsr process-failures switchover
```

# service tcp-small-servers

To enable small TCP servers such as the ECHO, use the **service tcp-small-servers** command in XR Config mode. To disable the TCP server, use the **no** form of this command.

**service** {**ipv4** | **ipv6**} **tcp-small-servers** [**max-servers** {*number* | **no-limit**}] [*access-list-name*]
**no service** {**ipv4** | **ipv6**} **tcp-small-servers** [**max-servers** {*number* | **no-limit**}] [*access-list-name*]

## Syntax Description

| | |
|---|---|
| **ip4** | Specifies IPv4 small servers. |
| **ipv6** | Specifies IPv6 small servers. |
| **max-servers** | (Optional) Sets the number of allowable TCP small servers. |
| *number* | (Optional) Number value. Range is 1 to 2147483647. |
| **no-limit** | (Optional) Sets no limit to the number of allowable TCP small servers. |
| *access-list-name* | (Optional) The name of an access list. |

## Command Default

TCP small servers are disabled.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The TCP small servers currently consist of three services: Discard (port 9), Echo (port 7), and Chargen (port 19). These services are used to test the TCP transport functionality. The Discard server receives data and discards it. The Echo server receives data and echoes the same data to the sending host. The Chargen server generates a sequence of data and sends it to the remote host.

## Task ID

| Task ID | Operations |
|---|---|
| ipv4 | read, write |
| ip-services | read, write |

## Examples

In the following example, small IPv4 TCP servers are enabled:

```
RP/0/RP0/CPU0:router(config)# service ipv4 tcp-small-servers max-servers 5 acl100
```

## Related Commands

| Command | Description |
|---|---|
| service udp-small-servers, on page 519 | Enables small UDP servers such as the ECHO. |
| show cinetd services | Displays the services whose processes are spawned by cinetd. |

# service udp-small-servers

To enable small User Datagram Protocol (UDP) servers such as the ECHO, use the **service udp-small-servers** command in XR Config mode. To disable the UDP server, use the **no** form of this command.

**service** {**ipv4** | **ipv6**} **udp-small-servers** [**max-servers** {*number* | **no-limit**}] [*access-list-name*]
**no service** {**ipv4** | **ipv6**} **udp-small-servers** [**max-servers** {*number* | **no-limit**}] [*access-list-name*]

## Syntax Description

| | |
|---|---|
| **ip4** | Specifies IPv4 small servers. |
| **ipv6** | Specifies IPv6 small servers. |
| **max-servers** | (Optional) Sets the number of allowable UDP small servers. |
| *number* | (Optional) Number value. Range is 1 to 2147483647. |
| **no-limit** | (Optional) Sets no limit to the number of allowable UDP small servers. |
| *access-list-name* | (Optional) Name of an access list. |

## Command Default

UDP small servers are disabled.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The UDP small servers currently consist of three services: Discard (port 9), Echo (port 7), and Chargen (port 19). These services are used to test the UDP transport functionality. The discard server receives data and discards it. The echo server receives data and echoes the same data to the sending host. The chargen server generates a sequence of data and sends it to the remote host.

## Task ID

| Task ID | Operations |
|---|---|
| ipv6 | read, write |
| ip-services | read, write |

## Examples

The following example shows how to enable small IPv6 UDP servers and set the maximum number of allowable small servers to 10:

```
RP/0/RP0/CPU0:router(config)# service ipv6 udp-small-servers max-servers 10
```

**Related Commands**

| Command | Description |
|---|---|
| service tcp-small-servers, on page 517 | Enables small TCP servers such as the ECHO. |

# show nsr ncd client

To display information about the clients for nonstop routing (NSR) Consumer Demuxer (NCD), use the **show nsr ncd client** command in XR EXEC mode.

**show nsr ncd client** {*PID value* | **all** | **brief**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *PID v alue* | Process ID (PID) information for a specific client. The range is from 0 to 4294967295. |
| **all** | Displays detailed information about all the clients. |
| **brief** | Displays brief information about all the clients. |
| **location** *node-id* | (Optional) Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows detailed information about all the clients:

```
RP/0/RP0/CPU0:router# show nsr ncd client all

Client PID                          : 3874979
Client Protocol                     : TCP
Client Instance                     : 1
Total packets received              : 28
Total acks received                 : 0
Total packets/acks accepted         : 28
Errors in changing packet ownership  : 0
Errors in setting application offset : 0
Errors in enqueuing to client        : 0
Time of last clear                  : Never cleared
```

The following sample output shows brief information about all the clients:

```
RP/0/RP0/CPU0:router# show nsr ncd client brief
```

```
                                Total   Total   Accepted
Pid    Protocol    Instance     Packets Acks    Packets/Acks
3874979   TCP           1          28  0              28
```

This table describes the significant fields shown in the display.

***Table 72: show nsr ncd client Command Field Descriptions***

| Field | Description |
|---|---|
| Client PID | Process ID of the client process. |
| Client Protocol | Protocol of the client process. The protocol can be either TCP, OSPF, or BGP. |
| Client Instance | Instance number of the client process. There can be more than one instance of a routing protocol, such as OSPF. |
| Total packets received | Total packets received from the partner stack on the partner route processor (RP). |
| Total acks received | Total acknowledgements received from the partner stack on the partner RP for the packets sent to the partner stack. |
| Total packets/acks accepted | Total packets and acknowledgements received from the partner stack on the partner RP. |
| Errors in changing packet ownership | NCD changes the ownership of the packet to that of the client before queueing the packet to the client. This counter tracks the errors, if any, in changing the ownership. |
| Errors in setting application offset | NCD sets the offset of the application data in the packet before queueing the packet to the client. This counter tracks the errors, if any, in setting this offset. |
| Errors in enqueuing to client | Counter tracks any queueing errors. |
| Time of last clear | Statistics last cleared by the user. |

## Related Commands

| Command | Description |
|---|---|
| clear nsr ncd client, on page 491 | Clears the counters for the NSR Consumer Demuxer (NCD) client. |
| clear nsr ncd queue, on page 493 | Clears the counters for the NSR Consumer Demuxer (NCD) queue. |
| show nsr ncd queue, on page 523 | Displays information about the nonstop routing (NSR) Consumer Queue and Dispatch (QAD) queues. |

# show nsr ncd queue

To display information about the queues that are used by the nonstop routing (NSR) applications to communicate with their partner stacks on the partner route processors (RPs), use the **show nsr ncd queue** command in XR EXEC mode.

**show nsr ncd queue** {**all** | **brief** | **high** | **low**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **all** | Displays detailed information about all the consumer queues. |
| **brief** | Displays brief information about all the consumer queues. |
| **high** | Displays information about high-priority Queue and Dispatch (QAD) queues. |
| **low** | Displays information about low-priority QAD queues. |
| **location** *node-id* | (Optional) Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows brief information about all the consumer queues:

```
RP/0/RP0/CPU0:router# show nsr ncd queue brief

                        Total        Accepted
          Queue         Packets       Packets
       NSR_LOW              992           992
      NSR_HIGH                0             0
```

This table describes the significant fields shown in the display.

***Table 73: show nsr ncd queue Command Field Descriptions***

| Field | Description |
|---|---|
| Total Packets | Total number of packets that are received from the partner stack. |

| Field | Description |
|---|---|
| Accepted Packets | Number of received packets that were accepted after performing some validation tasks. |
| Queue | Name of queue. NSR_HIGH and NSR_LOW are the two queues. High priority packets flow on the NSR_HIGH queue. Low priority packets flow on the NSR_LOW queue. |

## Related Commands

| Command | Description |
|---|---|
| clear nsr ncd client, on page 491 | Clears the counters for the NSR consumer demuxer (NCD) client. |
| clear nsr ncd queue, on page 493 | Clears the counters for the NSR consumer demuxer (NCD) queue. |
| show nsr ncd client, on page 521 | Displays information about the clients for NSR consumer demuxer(NCD). |

# show raw brief

To display information about active RAW IP sockets, use the **show raw brief** command in XR EXEC mode.

**show raw brief** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

Protocols such as Open Shortest Path First (OSPF) and Protocol Independent Multicast (PIM) use long-lived RAW IP sockets. The **ping** and **traceroute** commands use short-lived RAW IP sockets. Use the **show raw brief** command if you suspect a problem with one of these protocols.

**Task ID**

| Task ID | Operations |
|---|---|
| transport | read |

**Examples**

The following is sample output from the **show raw brief** command:

```
RP/0/RP0/CPU0:router# show raw brief

PCB      Recv-Q Send-Q Local Address          Foreign Address Protocol
0x805188c        0      0 0.0.0.0               0.0.0.0            2
0x8051dc8        0      0 0.0.0.0               0.0.0.0          103
0x8052250        0      0 0.0.0.0               0.0.0.0          255
```

This table describes the significant fields shown in the display.

***Table 74: show raw brief Command Field Descriptions***

| Field | Description |
|---|---|
| PCB | Protocol control block address. This is the address to a structure that contains connection information such as local address, foreign address, local port, foreign port, and so on. |
| Recv-Q | Number of bytes in the receive queue. |
| Send-Q | Number of bytes in the send queue. |
| Local Address | Local address and local port. |

| Field | Description |
|---|---|
| Foreign Address | Foreign address and foreign port. |
| Protocol | Protocol that is using the RAW IP socket. For example, the number 2 is IGMP, 103 is PIM, and 89 is OSPF. |

# show raw detail pcb

To display detailed information about active RAW IP sockets, use the **show raw detail pcb** command in XR EXEC mode.

**show raw detail pcb** {*pcb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *pcb-address* | Displays statistics for a specified RAW connection. |
| **all** | Displays statistics for all RAW connections. |
| **location** *node-id* | (Optional) Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **show raw detail pcb** command displays detailed information for all connections that use the RAW transport. Information that is displayed includes family type (for example, 2 for AF_INET also known as IPv4), PCB address, Layer 4 (also known as transport) protocol, local address, foreign address, and any filter that is being used.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following is sample output from the **show raw detail pcb** command:

```
RP/0/RP0/CPU0:router# show raw detail pcb 0x807e89c

============================================================
PCB is 0x807e89c, Family: 2, PROTO: 89
 Local host: 0.0.0.0
 Foreign host: 0.0.0.0

Current send queue size: 0
Current receive queue size: 0
Paw socket: Yes
```

This table describes the significant fields shown in the display.

***Table 75: show raw detail pcb Command Field Descriptions***

| Field | Description |
|---|---|
| JID | Job ID of the process that created the socket. |
| Family | Network protocol. IPv4 is 2; IPv6 is 26. |
| PCB | Protocol control block address. |
| L4-proto | Layer 4 (also known as transport) protocol. |
| Laddr | Local address. |
| Faddr | Foreign address. |
| ICMP error filter mask | If an ICMP filter is being set, output in this field has a nonzero value. |
| LPTS socket options | If an LPTS option is being set, output in this field has a nonzero value. |
| Packet Type Filters | Packet filters that are being set for a particular RAW socket, including the number of packets for that filter type. Multiple filters can be set. |

# show raw extended-filters

To display information about active RAW IP sockets, use the **show raw extended-filters** command in XR EXEC mode.

**show raw extended-filters** {**interface-filter location** *node-id* | **location** *node-id* | **paktype-filter location** *node-id*}

## Syntax Description

| | |
|---|---|
| **interface-filter** | Displays the protocol control blocks (PCBs) with configured interface filters. |
| **location** *node-id* | Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **paktype-filter** | Displays the PCBs with configured packet type filters. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **show raw extended-filters** command displays detailed information for all connections that use the RAW transport. Information that is displayed includes family type (for example, 2 for AF_INET also known as IPv4), PCB address, Layer 4 (also known as transport) protocol, local address, foreign address, and any filter that is being used.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following is sample output from the **show raw extended-filters** command:

```
RP/0/RP0/CPU0:router# show raw extended-filters 0/0/CPU0

Total Number of matching PCB's in database: 1
JID: 0/0
Family: 2
PCB: 0x0803dd38
L4-proto: 1
Laddr: 0.0.0.0
Faddr: 0.0.0.0
ICMP error filter mask: 0x3ff
LPTS socket options: 0x0020
Packet Type Filters:
0
[220 pkts in]
3
[0 pkts in]
```

```
4
[0 pkts in]
```

This table describes the significant fields shown in the display.

***Table 76: show raw extended-filters Output Command Field Descriptions***

| Field | Description |
|---|---|
| JID | Job ID of the process that created the socket. |
| Family | Network protocol. IPv4 is 2; IPv6 is 26. |
| PCB | Protocol control block address. |
| L4-proto | Layer 4 (also known as transport) protocol. |
| Laddr | Local address. |
| Faddr | Foreign address. |
| ICMP error filter mask | If an ICMP filter is being set, output in this field has a nonzero value. |
| LPTS socket options | If an LPTS option is being set, output in this field has a nonzero value. |
| Packet Type Filters | Packet filters that are being set for a particular RAW socket, including the number of packets for that filter type. Multiple filters can be set. |

# show raw statistics pcb

To display statistics for a single RAW connection or for all RAW connections, use the **show raw statistics pcb** command in XR EXEC mode.

**show raw statistics pcb** {**all** | **pcb-address**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **all** | Displays statistics for all RAW connections. |
| **pcb-address** | Displays statistics for a specified RAW connection. |
| **location** *node-id* | (Optional) Displays RAW statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **all** keyword to display all RAW connections. If a specific RAW connection is desired, then enter the protocol control block (PCB) address of that RAW connection. Use the **show raw brief** command to obtain the PCB address.

Use the **location** keyword and *node-id* argument to display RAW statistics for a designated node.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

In the following example, statistics for a RAW connection with PCB address 0x80553b0 are displayed:

```
RP/0/RP0/CPU0:router# show raw statistics pcb 0x80553b0

Statistics for PCB 0x80553b0
Send: 0 packets received from application
0 xipc pulse received from application
0 packets sent to network
0 packets failed getting queued to network
Rcvd: 0 packets received from network
0 packets queued to application
0 packets failed queued to application
```

In this example, statistics for all RAW connections are displayed:

```
RP/0/RP0/CPU0:router# show raw statistics pcb all
```

```
Statistics for PCB 0x805484c
Send: 0 packets received from application
0 xipc pulse received from application
0 packets sent to network
0 packets failed getting queued to network
Rcvd: 0 packets received from network
0 packets queued to application
0 packets failed queued to application
```

This table describes the significant fields shown in the display.

***Table 77: show raw statistics pcb Command Field Descriptions***

| Field | Description |
|---|---|
| Send: | Statistics in this section refer to packets sent from an application to RAW. |
| xipc pulse received from application | Number of notifications sent from applications to RAW. |
| packets sent to network | Number of packets sent to the network. |
| packets failed getting queued to network | Number of packets that failed to get queued to the network. |
| Rcvd: | Statistics in this section refer to packets received from the network. |
| packets queued to application | Number of packets queued to an application. |
| packets failed queued to application | Number of packets that failed to get queued to an application. |

## Related Commands

| Command | Description |
|---|---|
| clear raw statistics pcb, on page 495 | Clears statistics for either a single RAW connection or for all RAW connections. |
| show raw brief, on page 525 | Displays information about active RAW IP sockets. |

# show tcp brief

To display a summary of the TCP connection table, use the **show tcp brief** command in XR EXEC mode.

**show tcp brief** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operations |
|---|---|
| transport | read |

**Examples**

The following is sample output from the **show tcp brief** command:

```
RP/0/RP0/CPU0:router# show tcp brief

TCPCB     Recv-Q Send-Q  Local Address          Foreign Address        State
0x80572a8      0      0  0.0.0.0:513            0.0.0.0:0              LISTEN
0x8056948      0      0  0.0.0.0:23             0.0.0.0:0              LISTEN
0x8057b60      0      3  10.8.8.2:23            10.8.8.1:1025          ESTAB
```

This table describes the significant fields shown in the display.

***Table 78: show tcp brief Command Field Descriptions***

| Field | Description |
|---|---|
| TCPCB | Memory address of the TCP control block. |
| Recv-Q | Number of bytes waiting to be read. |
| Send-Q | Number of bytes waiting to be sent. |
| Local Address | Source address and port number of the packet. |
| Foreign Address | Destination address and port number of the packet. |

| Field | Description |
|---|---|
| State | State of the TCP connection. |

## Related Commands

| Command | Description |
|---|---|
| clear tcp pcb, on page 511 | Clears the TCP connection. |

# show tcp detail

To display the details of the TCP connection table, use the **show tcp detail** command in XR EXEC mode.

**show tcp detail pcb** [{*value* | **all**}]

## Syntax Description

| | |
|---|---|
| **pcb** | Displays TCP connection information. |
| *value* | Displays a specific connection information. Range is from 0 to ffffffff. |
| **all** | Displays all connections information. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following is sample output from the **show tcp detail pcb all** command:

```
RP/0/RP0/CPU0:router# show tcp detail pcb all

Connection state is LISTEN, I/O status: 0, socket status: 0
PCB 0x8092774
Local host: 0.0.0.0, Local port: 23
Foreign host: 0.0.0.0, Foreign port: 0

Current send queue size: 0 (max 16384)
Current receive queue size: 0 (max 16384)  mis-ordered: 0 bytes

Timer          Starts    Wakeups         Next(msec)
Retrans             0          0                 0
SendWnd             0          0                 0
TimeWait            0          0                 0
AckHold             0          0                 0
KeepAlive           0          0                 0
PmtuAger            0          0                 0
GiveUp              0          0                 0
Throttle            0          0                 0
iss: 0           snduna: 0           sndnxt: 0
sndmax: 0           sndwnd: 0           sndcwnd: 1073725440
   irs: 0           rcvnxt: 0           rcvwnd: 16384   rcvadv: 0
```

# show tcp extended-filters

To display the details of the TCP extended-filters, use the **show tcp extended-filters** command in XR EXEC mode.

**show tcp extended-filters** [**location** *node-id*]
**peer-filter** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **peer-filter** | Displays connections with peer filter configured. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following is sample output from the **show tcp extended-filters** command for a specific location (0/0/CPU0):

```
RP/0/RP0/CPU0:router# show tcp extended-filters location 0/0/CPU0

Total Number of matching PCB's in database: 3
-----------------------------------
JID: 135
Family: 2
PCB: 0x4826c5dc
L4-proto: 6
Lport: 23
Fport: 0
Laddr: 0.0.0.0
Faddr: 0.0.0.0
ICMP error filter mask: 0x12
LPTS options: 0x00000000
-----------------------------------

-----------------------------------
JID: 135
Family: 2

PCB: 0x4826dd8c
```

```
L4-proto: 6
Lport: 23
Fport: 59162
Laddr: 12.31.22.10
Faddr: 223.255.254.254
ICMP error filter mask: 0x12
LPTS options: 0x00000000
-----------------------------------

-----------------------------------
JID: 135
Family: 2
PCB: 0x4826cac0
L4-proto: 6
Lport: 23
Fport: 59307
Laddr: 12.31.22.10
Faddr: 223.255.254.254
ICMP error filter mask: 0x12
LPTS options: 0x00000000
-----------------------------------
```

# show tcp statistics

To display TCP statistics, use the **show tcp statistics** command in XR EXEC mode.

**show tcp statistics** [{**pcb** {**all***pcb-address*} | **summary** | **clients**}] [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **pcb** *pcb-address* | (Optional) Displays detailed statistics for a specified connection. |
| **pcb all** | (Optional) Displays detailed statistics for all connections. |
| **summary** | (Optional) Clears summary statistic for a specific node or connection. |
| **location** *node-id* | (Optional) Displays statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **clients** | (Optional) Displays detailed statistics for all clients. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following is sample output from the **show tcp statistics** command:

```
RP/0/RP0/CPU0:router# show tcp statistics pcb 0x08091bc8

Statistics for PCB 0x8091bc8
Send:   0 bytes received from application
        0 xipc pulse received from application
        0 bytes sent to network
        0 packets failed getting queued to network
Rcvd:   0 packets received from network
        0 packets queued to application
        0 packets failed queued to application
```

This table describes the significant fields shown in the display.

*Table 79: show tcp statistics Command Field Descriptions*

| Field | Description |
|---|---|
| Send | Statistics in this section refer to packets sent by the router. |
| Rcvd: | Statistics in this section refer to packets received by the router. |

## Related Commands

| Command | Description |
|---|---|
| clear tcp statistics, on page 512 | Clears TCP statistics. |

# show tcp nsr brief

To display the key nonstop routing (NSR) state of TCP connections on different nodes, use the **show tcp nsr brief** command in XR EXEC mode.

**show tcp nsr brief** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays information for all TCP sessions for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

If a value is not specified, the current RP in which the command is being executed is taken as the location.

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

The **location** keyword is used so that active and standby TCP instances are independently queried.

**Task ID**

| Task ID | Operations |
|---|---|
| transport | read |

**Examples**

The following sample output shows the administrative and operational NSR state of each TCP session in the NSR column:

```
RP/0/RP0/CPU0:router# show tcp nsr brief

PCB        Local Address          Foreign Address        NSR   RcvOnly
0x482c6b8c
5.1.1.1:646
5.1.1.2:23945          Down  No
0x482db564
5.1.1.1:646
5.1.1.2:25398          Down  No
0x482844e0
5.1.1.1:646
5.1.1.2:25430          Down  No
0x482c9284
5.1.1.1:646
5.1.1.2:37434          Down  No
0x482d98c8
5.1.1.1:646
5.1.1.2:37895          Down  No
0x482d6018
5.1.1.1:646
5.1.1.2:50616          Down  No
0x482c7f08
5.1.1.1:646
5.1.1.2:55860          Down  No
```

```
0x482dbab0
5.1.1.1:646
5.1.1.2:56656          Down  No
0x482d7394
5.1.1.1:646
5.1.1.2:57365          Down  No
0x482d854c
5.1.1.1:646
5.1.1.2:59927          Down  No
```

This table describes the significant fields shown in the display.

***Table 80: show tcp nsr brief Command Field Descriptions***

| Field | Description |
|---|---|
| PCB | Protocol Control Block (PCB). |
| Local Address | Local address and port of the TCP connection. |
| Foreign Address | Foreign address and port of the TCP connection. |
| NSR | Current operational NSR state of this TCP connection. |
| RevOnly | If yes, the TCP connection is replicated only in the receive direction. Some applications may need to replicate a TCP connection that is only in the receive direction. |

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr pcb, on page 499 | Brings the NSR down on a specified connection or all connections. |
| show tcp nsr client brief, on page 542 | Displays brief information about the state of nonstop routing (NSR) for the TCP clients on different nodes. |

# show tcp nsr client brief

To display brief information about the state of nonstop routing (NSR) for TCP clients on different nodes, use the **show tcp nsr client brief** command in XR EXEC mode.

**show tcp nsr client brief** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays brief client information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

If a value is not specified, the current RP in which the command is being executed is taken as the location.

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

The **location** keyword is used so that active and standby TCP instances are independently queried.

**Task ID**

| Task ID | Operations |
|---|---|
| transport | read |

**Examples**

The following sample output is from the **show tcp nsr client brief** command:

```
RP/0/RP0/CPU0:router# show tcp nsr client brief location 0/1/CPU0


CCB          Proc Name     Instance Sets Sessions/NSR Up Sessions
0x482bf378   mpls_ldp  1        1             1/1
0x482bd32c   mpls_ldp  2        1             0/0
```

This table describes the significant fields shown in the display.

***Table 81: show tcp nsr client brief Command Field Descriptions***

| Field | Description |
|---|---|
| CCB | Client Control Block (CCB). Unique ID to identify the client. |
| Proc Name | Name of the client process. |
| Instance | Instance is identified as the instance number of the client process because there can be more than one instance for a routing application. |
| Sets | Set number is identified as the ID of the session-set. |
| Sessions/NSR Up Sessions | Total sessions in the set versus the number of the sessions in which NSR is up. |

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr client, on page 497 | Clears detailed information about the nonstop routing (NSR) clients. |
| show tcp nsr brief, on page 540 | Displays the key nonstop routing (NSR) state of TCP connections on different nodes. |

# show tcp nsr detail client

To display detailed information about the nonstop routing (NSR) clients, use the **show tcp nsr detail client** command in XR EXEC mode.

**show tcp nsr detail client** {*ccb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *ccb-address* | Client Control Block (CCB) address range for the specific client information. 0 to ffffffff. For example, the address range can be 0x482a4e20. |
| **all** | Specifies all the clients. |
| **location** *node-id* | (Optional) Displays client information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows detailed information for all clients:

```
RP/0/RP0/CPU0:router# show tcp nsr detail client all


============================================================
CCB 0x482b25d8, Proc Name mpls_ldp
Instance ID 1, Job ID 360
Number of session-sets 2
Number of sessions 3
Number of NSR Synced sessions 1
Connected at: Sun Jun 10 07:05:31 2007
Registered for notifications: Yes

============================================================
CCB 0x4827fd30, Proc Name mpls_ldp
Instance ID 2, Job ID 361
Number of session-sets 1
Number of sessions 2
Number of NSR Synced sessions 2
Connected at: Sun Jun 10 07:05:54 2007
Registered for notifications: Yes
============================================================

RP/0/RP0/CPU0:router# show tcp nsr detail client all location 1
RP/0/RP0/CPU0:router# show tcp nsr detail client all location 0/1/CPU0

============================================================
```

```
CCB 0x482bf378, Proc Name mpls_ldp
Instance ID 1, Job ID 360
Number of session-sets 1
Number of sessions 1
Number of NSR Synced sessions 1
Connected at: Sun Jun 10 07:05:41 2007
Registered for notifications: Yes

============================================================
CCB 0x482bd32c, Proc Name mpls_ldp
Instance ID 2, Job ID 361
Number of session-sets 1
Number of sessions 2
Number of NSR Synced sessions 2
Connected at: Sun Jun 10 07:06:01 2007
Registered for notifications: Yes
```

## Related Commands

| Command | Description |
|---|---|
| show tcp nsr detail pcb, on page 546 | Displays detailed information about the nonstop routing (NSR) state of TCP connections. |
| show tcp nsr detail session-set, on page 549 | Displays the detailed information about the nonstop routing (NSR) state of the session sets on different nodes. |

# show tcp nsr detail pcb

To display detailed information about the nonstop routing (NSR) state of TCP connections, use the **show tcp nsr detail pcb** command in XR EXEC mode.

**show tcp nsr detail pcb** {*pcb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *pcb-address* | PCB address range for the specific connection information. 0 to ffffffff. For example, the address range can be 0x482c6b8c. |
| **all** | Specifies all the connections. |
| **location** *node-id* | (Optional) Displays connection information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows the complete details for NSR for all locations:

```
RP/0/RP0/CPU0:router# show tcp nsr detail pcb all location 0/0/cpu0


==============================================================
PCB 0x482b6b0c, Client PID: 2810078
Local host: 5.1.1.1, Local port: 646
Foreign host: 5.1.1.2, Foreign port: 31466
SSCB 0x482bc80c, Client PID 2810078
Node Role: Active, Protected by: 0/1/CPU0, Cookie: 0x00001000

NSR State: Up, Rcv Path Replication only: No
Replicated to standby: Yes
Synchronized with standby: Yes
FSSN: 3005097735, FSSN Offset: 0

Sequence number of last or current initial sync: 1181461961
Initial sync started at: Sun Jun 10 07:52:41 2007
Initial sync ended   at: Sun Jun 10 07:52:41 2007

Number of incoming packets currently held: 1
```

```
          Pak#    SeqNum     Len     AckNum
         -----  ----------  -----  ----------
             1  3005097735      0  1172387202

Number of iACKS currently held: 0

==============================================================
PCB 0x482c2920, Client PID: 2810078
Local host: 5.1.1.1, Local port: 646
Foreign host: 5.1.1.2, Foreign port: 11229
SSCB 0x482bb3bc, Client PID 2810078
Node Role: Active, Protected by: 0/1/CPU0, Cookie: 0x00001000

NSR State: Down, Rcv Path Replication only: No
Replicated to standby: No
Synchronized with standby: No
NSR-Down Reason: Initial sync was aborted
NSR went down at: Sun Jun 10 11:55:38 2007

Initial sync in progress: No
Sequence number of last or current initial sync: 1181476338
Initial sync error, if any: 'ip-tcp' detected the 'warning' condition 'Initial sync operation
 timed out'
Source of initial sync error: Local TCP
Initial sync started at: Sun Jun 10 11:52:18 2007
Initial sync ended   at: Sun Jun 10 11:55:38 2007

Number of incoming packets currently held: 0

Number of iACKS currently held: 0

==============================================================
PCB 0x482baea0, Client PID: 2810078
Local host: 5.1.1.1, Local port: 646
Foreign host: 5.1.1.2, Foreign port: 41149
SSCB 0x482bb3bc, Client PID 2810078
Node Role: Active, Protected by: 0/1/CPU0, Cookie: 0x00001000

NSR State: Down, Rcv Path Replication only: No
Replicated to standby: No
Synchronized with standby: No
NSR-Down Reason: Initial sync was aborted
NSR went down at: Sun Jun 10 11:55:38 2007

Initial sync in progress: No
Sequence number of last or current initial sync: 1181476338
Initial sync error, if any: 'ip-tcp' detected the 'warning' condition 'Initial sync operation
 timed out'
Source of initial sync error: Local TCP
Initial sync started at: Sun Jun 10 11:52:18 2007
Initial sync ended   at: Sun Jun 10 11:55:38 2007

Number of incoming packets currently held: 0

Number of iACKS currently held: 0

==============================================================
PCB 0x482c35ac,  Client PID: 2859233
Local host: 5:1::1, Local port: 8889
Foreign host: 5:1::2, Foreign port: 14008
SSCB 0x4827fea8, Client PID 2859233
Node Role: Active, Protected by: 0/1/CPU0, Cookie: 0x0000001c

NSR State: Up, Rcv Path Replication only: No
```

```
Replicated to standby: Yes
Synchronized with standby: Yes
FSSN: 2962722865, FSSN Offset: 0

Sequence number of last or current initial sync: 1181474373
Initial sync started at: Sun Jun 10 11:19:33 2007
Initial sync ended   at: Sun Jun 10 11:19:33 2007

Number of incoming packets currently held: 0

Number of iACKS currently held: 0

============================================================
PCB 0x482c2f10,  Client PID: 2859233
Local host: 5:1::1, Local port: 8889
Foreign host: 5:1::2, Foreign port: 40522
SSCB 0x4827fea8, Client PID 2859233
Node Role: Active, Protected by: 0/1/CPU0, Cookie: 0x0000001b

NSR State: Up, Rcv Path Replication only: No
Replicated to standby: Yes
Synchronized with standby: Yes
FSSN: 3477316401, FSSN Offset: 0

Sequence number of last or current initial sync: 1181474373
Initial sync started at: Sun Jun 10 11:19:33 2007
Initial sync ended   at: Sun Jun 10 11:19:33 2007

Number of incoming packets currently held: 0

Number of iACKS currently held: 0
```

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr pcb, on page 499 | Brings the NSR down on a specified connection or all connection. |
| show tcp nsr detail client, on page 544 | Displays detailed information about the nonstop routing (NSR) clients. |
| show tcp nsr detail session-set, on page 549 | Displays the detailed information about the nonstop routing (NSR) state of the session sets on different nodes. |

# show tcp nsr detail session-set

To display the detailed information about the nonstop routing (NSR) state of the session sets on different nodes, use the **show tcp nsr detail session-set** command in XR EXEC mode.

**show tcp nsr detail session-set** {*sscb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *sscb-address* | Session-Set Control Block (SSCB) address range for the specific session set information. 0 to ffffffff. For example, the address range can be 0x482c6b8c. |
| **all** | Specifies all the session sets. |
| **location** *node-id* | (Optional) Displays information for session sets for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows all the session sets:

```
RP/0/RP0/CPU0:router# show tcp nsr detail session-set all


============================================================
SSCB 0x482bc80c, Client PID: 2810078
Set Id: 1, Addr Family: IPv4
Role: Active, Protected by: 0/1/CPU0, Well known port: 646
Sessions: total 1, synchronized 1
Initial sync in progress: No
        Sequence number of last or current initial sync: 1181461961
        Number of sessions in the initial sync: 1
        Number of sessions already synced: 1
        Number of sessions that failed to sync: 0
        Initial sync started at: Sun Jun 10 07:52:41 2007
        Initial sync ended   at: Sun Jun 10 07:52:41 2007

============================================================
SSCB 0x482bb3bc, Client PID: 2810078
Set Id: 2, Addr Family: IPv4
Role: Active, Protected by: 0/1/CPU0, Well known port: 646
```

```
Sessions: total 2, synchronized 0
Initial sync in progress: Yes
        Sequence number of last or current initial sync: 1181476338
        Initial sync timer expires in 438517602 msec
        Number of sessions in the initial sync: 2
        Number of sessions already synced: 0
        Number of sessions that failed to sync: 0
        Initial sync started at: Sun Jun 10 11:52:18 2007

============================================================
SSCB 0x4827fea8, Client PID: 2859233
Set Id: 1, Addr Family: IPv6
Role: Active, Protected by: 0/1/CPU0, Well known port: 8889
Sessions: total 2, synchronized 2
Initial sync in progress: No
        Sequence number of last or current initial sync: 1181474373
        Number of sessions in the initial sync: 2
        Number of sessions already synced: 2
        Number of sessions that failed to sync: 0
        Initial sync started at: Sun Jun 10 11:19:33 2007
        Initial sync ended   at: Sun Jun 10 11:19:33 2007
```

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr session-set, on page 502 | Clears information about session sets. |
| show tcp nsr detail client, on page 544 | Displays detailed information about the nonstop routing (NSR) clients. |
| show tcp nsr detail pcb, on page 546 | Displays detailed information about the nonstop routing (NSR) state of TCP connections. |

# show tcp nsr session-set brief

To display brief information about the session sets for the nonstop routing (NSR) state on different nodes, use the **show tcp nsr session-set brief** command in XR EXEC mode.

**show tcp nsr session-set brief** [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| **location** *node-id* | (Optional) Displays information for session sets for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

A session set consists of a subset of the application's session in which the subset is protected by only one standby node. The TCP NSR state machine operates with respect to these session sets.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows all the session sets that are known to the TCP instance:

```
RP/0/RP0/CPU0:router# show tcp nsr session-set brief

SSCB       Client   LocalAPP Set-Id   Family    Role      Protect-Node Total/Synced
0x482bc80c 2810078  mpls_ldp#1    1     IPv4    Active     0/1/CPU0      1/1
0x482bb3bc 2810078  mpls_ldp#1    2     IPv4    Active     0/1/CPU0      2/0
0x4827fea8 2859233  mpls_ldp#2    1     IPv6    Active     0/1/CPU0      2/2
```

The following sample output shows brief information about the session sets for location 0/1/CPU0:

```
RP/0/RP0/CPU0:router# show tcp nsr session-set brief location 0/1/CPU0

SSCB           Client   LocalAPP  Set-Id      Family   Role   Protect-Node  Total/Synced
0x4827ff74     602319   mpls_ldp#1     1       IPv4    Stdby  0/0/CPU0      1/1
0x482b8f54     602320   mpls_ldp#2     1       IPv6    Stdby  0/0/CPU0      2/2
```

This table describes the significant fields shown in the display.

***Table 82: show tcp nsr session-set brief Command Field Descriptions***

| Field | Description |
|---|---|
| SSCB | Unique ID for Session-Set Control Block (SSCB) to identify a session-set of a client. |
| Client | PID of the client process. |
| LocalAPP | Name and instance number of the client process. |
| Set-Id | ID of the session-set. |
| Family | Address family of the sessions added to the session set for IPv4 or IPv6. |
| Role | Role of the TCP stack for active or standby. |
| Protect-Node | Node that is offering the protection, for example, partner node. |
| Total/Synced | Total number of sessions in the set versus the sessions that have been synchronized. |

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr session-set, on page 502 | Clears information about session sets. |
| show tcp nsr detail session-set, on page 549 | Displays the detailed information about the nonstop routing (NSR) state of the session sets on different nodes. |

# show tcp nsr statistics client

To display the nonstop routing (NSR) statistics for the clients, use the **show tcp nsr statistics client** command in XR EXEC mode.

**show tcp nsr statistics client** {*ccb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *ccb-address* | Client Control Block (CCB) address range for the specific statistics information for the client. 0 to ffffffff. For example, the address range can be 0x482c6b8c. |
| **all** | Specifies all the statistics for the clients. |
| **location** *node-id* | (Optional) Displays statistics for the client for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows all the statistics for the client:

```
RP/0/RP0/CPU0:router# show tcp nsr statistics client all


============================================================
CCB: 0x482b25d8
Name: mpls_ldp, Job ID: 360
Connected at: Thu Jan  1 00:00:00 1970

Notification Stats      : Queued  Failed  Delivered  Dropped
Init-Sync Done          :      0       0          0       0
Replicated Session Ready:      0       0          0       0
Operational Down        :      0       0          0       0
Last clear at: Sun Jun 10 12:19:12 2007


============================================================
CCB: 0x4827fd30
Name: mpls_ldp, Job ID: 361
Connected at: Sun Jun 10 07:05:54 2007
```

```
Notification Stats      : Queued  Failed  Delivered  Dropped
Init-Sync Done          :      1       0          1       0
Replicated Session Ready:      0       0          0       0
Operational Down        :      0       0          0       0
Last clear at: Never Cleared
```

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr statistics client, on page 504 | Clears the nonstop routing (NSR) statistics of the client. |
| show tcp nsr statistics pcb, on page 555 | Displays the nonstop routing (NSR) statistics for a given Protocol Control Block (PCB). |
| show tcp nsr statistics session-set, on page 557 | Displays the nonstop routing (NSR) statistics for a session set. |
| show tcp nsr statistics summary, on page 559 | Displays the nonstop routing (NSR) summary statistics across all TCP sessions. |

# show tcp nsr statistics pcb

To display the nonstop routing (NSR) statistics for a given Protocol Control Block (PCB), use the **show tcp nsr statistics pcb** command in XR EXEC mode.

**show tcp nsr statistics pcb** {*pcb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *pcb-address* | PCB address range for the specific connection information. 0 to ffffffff. For example, the address range can be 0x482c6b8c. |
| **all** | Specifies all the connection statistics. |
| **location** *node-id* | (Optional) Displays connection statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows all NSR statistics:

```
RP/0/RP0/CPU0:router# show tcp nsr statistics pcb all

==============================================================
PCB 0x482b6b0c
Number of times NSR went up: 0
Number of times NSR went down: 0
Number of times NSR was disabled: 0
Number of times fail-over occured : 0
Last clear at: Sun Jun 10 13:55:35 2007


==============================================================
PCB 0x482c2920
Number of times NSR went up: 2
Number of times NSR went down: 2
Number of times NSR was disabled: 0
Number of times fail-over occured : 0
Last clear at: Never Cleared
```

```
==============================================================
PCB 0x482baea0
Number of times NSR went up: 2
Number of times NSR went down: 2
Number of times NSR was disabled: 0
Number of times fail-over occured : 0
Last clear at: Never Cleared

==============================================================
PCB 0x482c35ac
Number of times NSR went up: 4
Number of times NSR went down: 2
Number of times NSR was disabled: 1
Number of times fail-over occured : 0
Last clear at: Never Cleared

==============================================================
PCB 0x482c2f10
Number of times NSR went up: 4
Number of times NSR went down: 2
Number of times NSR was disabled: 1
Number of times fail-over occured : 0
Last clear at: Never Cleared
```

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr statistics pcb, on page 506 | Clears the nonstop routing (NSR) statistics for TCP connections. |
| show tcp nsr statistics client, on page 553 | Displays the nonstop routing (NSR) statistics for the clients. |
| show tcp nsr statistics session-set, on page 557 | Displays the nonstop routing (NSR) statistics for a session set. |
| show tcp nsr statistics summary, on page 559 | Displays the nonstop routing (NSR) summary statistics across all TCP sessions. |

# show tcp nsr statistics session-set

To display the nonstop routing (NSR) statistics for a session set, use the **show tcp nsr statistics session-set** command in XR EXEC mode.

**show tcp nsr statistics session-set** {*sscb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *sscb-address* | Session-Set Control Block (SSCB) address range for the specific session set information for the statistics. 0 to ffffffff. For example, the address range can be 0x482b3444. |
| **all** | Specifies all the session sets for the statistics. |
| **location** *node-id* | (Optional) Displays session set information for the statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

If a value is not specified, the current RP in which the command is being executed is taken as the location.

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **location** keyword is used so that active and standby TCP instances are independently queried.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following sample output shows all session set information for the statistics:

```
RP/0/RP0/CPU0:router# show tcp nsr statistics session-set all


==================Session Set Stats ============================
SSCB 0x482bc80c, Set ID: 1
Number of times init-sync was attempted  :1
Number of times init-sync was successful :1
Number of times init-sync failed         :0
Number of times switch-over occured      :0
Last clear at: Never Cleared

==================Session Set Stats ============================
SSCB 0x482bb3bc, Set ID: 2
Number of times init-sync was attempted  :1
Number of times init-sync was successful :0
Number of times init-sync failed         :1
Number of times switch-over occured      :0
Last clear at: Never Cleared

==================Session Set Stats ============================
```

```
SSCB 0x4827fea8, Set ID: 1
Number of times init-sync was attempted  :0
Number of times init-sync was successful :0
Number of times init-sync failed         :0
Number of times switch-over occured      :0
Last clear at: Sun Jun 10 13:36:51 2007
```

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr statistics session-set, on page 508 | Clears the nonstop routing (NSR) statistics for session sets. |
| show tcp nsr statistics client, on page 553 | Displays the nonstop routing (NSR) statistics for the clients. |
| show tcp nsr statistics pcb, on page 555 | Displays the nonstop routing (NSR) statistics for a given Protocol Control Block (PCB). |
| show tcp nsr statistics summary, on page 559 | Displays the nonstop routing (NSR) summary statistics across all TCP sessions. |

# show tcp nsr statistics summary

To display the nonstop routing (NSR) summary statistics across all TCP sessions, use the **show tcp nsr statistics summary** command in XR EXEC mode.

**show tcp nsr statistics summary** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | (Optional) Displays information for the summary statistics for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

If a value is not specified, the current RP in which the command is being executed is taken as the location.

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

The **location** keyword is used so that active and standby TCP instances are independently queried.

**Task ID**

| Task ID | Operations |
|---|---|
| transport | read |

**Examples**

The following sample output shows the summary statistics for all TCP sessions:

```
RP/0/RP0/CPU0:router# show tcp nsr statistics summary


====================Summary Stats=========================
The last clear at Thu Jan  1 00:00:00 1970

Notif Statistic:
                         Queued  Failed  Delivered  Dropped
Init-sync Done          :      3       0          3        0
Replicated Session Ready:      0       0          0        0
Operational Down        :      8       0          8        0
QAD Msg Statistic:
Number of dropped messages from partner TCP stack(s)        : 0
Number of unknown messages from partner TCP stack(s)        : 0
Number of messages accepted from partner TCP stack(s)       : 31
Number of messages sent to partner TCP stack(s)             : 0
Number of messages failed to be sent to partner TCP stack(s): 0
IACK RX Msg Statistic:
Number of iACKs dropped because there is no PCB             : 0
Number of iACKs dropped because there is no datapath SCB    : 0
Number of iACKs dropped because SSO is not up               : 0
Number of stale iACKs dropped                               : 6
Number of iACKs not held because of an immediate match      : 0
Number of held packets dropped because of errors            : 0
```

## Related Commands

| Command | Description |
|---|---|
| clear tcp nsr statistics summary, on page 510 | Clears the statistics summary. |
| show tcp nsr statistics client, on page 553 | Displays the nonstop routing (NSR) statistics for the clients. |
| show tcp nsr statistics pcb, on page 555 | Displays the nonstop routing (NSR) statistics for a given Protocol Control Block (PCB). |
| show tcp nsr statistics session-set, on page 557 | Displays the nonstop routing (NSR) statistics for a session set. |

# show udp brief

To display a summary of the User Datagram Protocol (UDP) connection table, use the **show udp brief** command in XR EXEC mode.

**show udp brief** [**location** *node-id*]

**Syntax Description**

| | |
|---|---|
| **location** *node-id* | Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

**Command Default**

No default behavior or values

**Command Modes**

XR EXEC mode

**Command History**

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

**Usage Guidelines**

No specific guidelines impact the use of this command.

**Task ID**

| Task ID | Operations |
|---|---|
| transport | read |

**Examples**

The following is sample output from the **show udp brief** command:

```
RP/0/RP0/CPU0:router# show udp brief

PCB        Recv-Q Send-Q  Local Address          Foreign Address
0x8040c4c      0      0  0.0.0.0:7              0.0.0.0:0
0x805a120      0      0  0.0.0.0:9              0.0.0.0:0
0x805a430      0      0  0.0.0.0:19             0.0.0.0:0
0x805a740      0      0  0.0.0.0:67             0.0.0.0:0
0x804fcb0      0      0  0.0.0.0:123            0.0.0.0:0
```

This table describes the significant fields shown in the display.

***Table 83: show udp brief Command Field Descriptions***

| Field | Description |
|---|---|
| PCB | Protocol control block address. This is the address to a structure that contains connection information such as local address, foreign address, local port, foreign port, and so on. |
| Recv-Q | Number of bytes in the receive queue. |
| Send-Q | Number of bytes in the send queue. |
| Local Address | Local address and local port. |

| Field | Description |
|---|---|
| Foreign Address | Foreign address and foreign port. |

## Related Commands

| Command | Description |
|---|---|
| show tcp brief, on page 533 | Displays details of TCP connections. |

# show udp detail pcb

To display detailed information of the User Datagram Protocol (UDP) connection table, use the **show udp detail pcb** command in XR EXEC mode.

**show udp detail pcb** {*pcb-address* | **all**} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| *pcb-address* | Address of a specified UDP connection. |
| **all** | Provides statistics for all UDP connections. |
| **location** *node-id* | Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following is sample output from the **show udp detail pcb all** command:

```
RP/0/RP0/CPU0:router# show udp detail pcb all location 0/3/CPU0

==============================================
PCB is 0x4822fea0, Family: 2,
 Local host: 0.0.0.0:3784
 Foreign host: 0.0.0.0:0

Current send queue size: 0
Current receive queue size: 0
==============================================
PCB is 0x4822d0e0, Family: 2,
 Local host: 0.0.0.0:3785
 Foreign host: 0.0.0.0:0

Current send queue size: 0
Current receive queue size: 0
```

This table describes the significant fields shown in the display.

***Table 84: show raw pcb Command Field Descriptions***

| Field | Description |
|---|---|
| PCB | Protocol control block address. |
| Family | Network protocol. IPv4 is 2; IPv6 is 26. |
| Local host | Local host address. |
| Foreign host | Foreign host address. |
| Current send queue size | Size of the send queue (in bytes). |
| Current receive queue size | Size of the receive queue (in bytes). |

# show udp extended-filters

To display the details of the UDP extended-filters, use the **show udp extended-filters** command in XR EXEC mode.

**show udp extended-filters** {**location** *node-id* | **peer-filter** {**location** *node-id}*}

## Syntax Description

| | |
|---|---|
| **location** *node-id* | Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **peer-filter** | Displays connections with peer filter configured. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following is sample output from the **show udp extended-filters** command for a specific location (0/0/CPU0):

```
RP/0/RP0/CPU0:router# show udp extended-filters location 0/0/CPU0

Total Number of matching PCB's in database: 1
----------------------------------
JID: 248
Family: 2
PCB: 0x48247e94
L4-proto: 17
Lport: 646
Fport: 0
Laddr: 0.0.0.0
Faddr: 0.0.0.0
ICMP error filter mask: 0x0
LPTS options: 0x00000000
----------------------------------
```

# show udp statistics

To display User Datagram Protocol (UDP) statistics, use the **show udp statistics** command in XR EXEC mode.

**show udp statistics** {**clients** | **summary** | **pcb** {*pcb-addressall*}} [**location** *node-id*]

## Syntax Description

| | |
|---|---|
| summary | Displays summary statistics. |
| **pcb** *pcb-address* | Displays detailed statistics for each connection. |
| **pcb** *all* | Displays detailed statistics for all connections. |
| **location** *node-id* | Displays information for the designated node. The *node-id* argument is entered in the *rack/slot/module* notation. |
| **clients** | Displays detailed statistics for all clients. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

UDP clones the received packets if there are multiple multicast applications that are interested in receiving those packets.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read |

## Examples

The following is sample output from the **show udp statistics summary** command:

```
RP/0/RP0/CPU0:router# show udp statistics summary

UDP statistics:
Rcvd: 0 Total, 0 drop, 0 no port
      0 checksum error, 0 too short
Sent: 0 Total, 0 error
0 Total forwarding broadcast packets
0 Cloned packets, 0 failed cloningication
```

This table describes the significant fields shown in the display.

*Table 85: show udp Command Field Descriptions*

| Field | Description |
|---|---|
| Rcvd: Total | Total number of packets received. |
| Rcvd: drop | Total number of packets received that were dropped. |
| Rcvd: no port | Total number of packets received that have no port. |
| Rcvd: checksum error | Total number of packets received that have a checksum error. |
| Rcvd: too short | Total number of packets received that are too short for UDP packets. |
| Sent: Total | Total number of packets sent successfully. |
| Sent: error | Total number of packets that cannot be sent due to errors. |
| Total forwarding broadcast packets | Total number of packets forwarded to the helper address. |
| Cloned packets | Total number of packets cloned successfully. |
| failed cloning | Total number of packets that failed cloning. |

## Related Commands

| Command | Description |
|---|---|
| clear udp statistics, on page 513 | Clears UDP statistics. |

# tcp mss

To configure the TCP maximum segment size that determines the size of the packet that TCP uses for sending data, use the **tcp mss** command in XR Config mode.

**tcp mss** *segment-size*

## Syntax Description

| | |
|---|---|
| segment-size | Size, in bytes, of the packet that TCP uses to send data. Range is 68 to 10000 bytes. |

## Command Default

If this configuration does not exist, TCP determines the maximum segment size based on the settings specified by the application process, interface maximum transfer unit (MTU), or MTU received from Path MTU Discovery.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read, write |

## Examples

This example shows how to configure the TCP maximum segment size:

```
RP/0/RP0/CPU0:router(config)# tcp mss 1460
RP/0/RP0/CPU0:router(config)# exit

Uncommitted changes found, commit them? [yes]:
RP/0/RP0/CPU0:router:Sep  8 18:29:51.084 : config[65700]: %LIBTARCFG-6-COMMIT :

Configuration committed by user 'lab'.   Use 'show commit changes 1000000596' to view the
changes.
RP/0/RP0/CPU0:routerSep  8 18:29:51.209 : config[65700]: %SYS-5-CONFIG_I : Configured from
 console by lab
```

# tcp path-mtu-discovery

To allow TCP to automatically detect the highest common maximum transfer unit (MTU) for a connection, use the **tcp path-mtu-discovery** in XR Config mode. To reset the default, use the **no** form of this command.

**tcp path-mtu-discovery** [{**age-timer** *minutes*}]
**no tcp path-mtu-discovery**

## Syntax Description

| | |
|---|---|
| **age-timer** *minutes* | (Optional) Specifies a value in minutes. Range is 10 to 30. |

## Command Default

Disabled

**age-timer** default is 10 minutes

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **tcp path-mtu-discovery** command to allow TCP to automatically detect the highest common MTU for a connection, such that when a packet traverses between the originating host and the destination host the packet is not fragmented and then reassembled.

The age timer value is in minutes, with a default value of 10 minutes. The age timer is used by TCP to automatically detect if there is an increase in MTU for a particular connection. If the **infinite** keyword is specified, the age timer is turned off.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read, write |

## Examples

The following example shows how to set the age timer to 20 minutes:

```
RP/0/RP0/CPU0:router(config)# tcp path-mtu-discovery age-timer 20
```

# tcp selective-ack

To enable TCP selective acknowledgment (ACK) and identify which segments in a TCP packet have been received by the remote TCP, use the **tcp selective-ack** command in XR Config mode. To reset the default, use the **no** form of this command.

**tcp selective-ack**
**no tcp selective-ack**

## Syntax Description

This command has no keywords or arguments.

## Command Default

TCP selective ACK is disabled.

## Command Modes

XR Config mode

## Usage Guidelines

If TCP Selective ACK is enabled, each packet contains information about which segments have been received by the remote TCP. The sender can then resend only those segments that are lost. If selective ACK is disabled, the sender receives no information about missing segments and automatically sends the first packet that is not acknowledged and then waits for the other TCP to respond with what is missing from the data stream. This method is inefficient in Long Fat Networks (LFN), such as high-speed satellite links in which the bandwidth * delay product is large and valuable bandwidth is wasted waiting for retransmission.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read, write |

## Examples

In the following example, the selective ACK is enabled:

```
RP/0/RP0/CPU0:router(config)# tcp selective-ack
```

## Related Commands

| Command | Description |
|---|---|
| tcp timestamp, on page 572 | Measures the round-trip time of a packet. |

# tcp synwait-time

To set a period of time the software waits while attempting to establish a TCP connection before it times out, use the **tcp synwait-time** command in XR Config mode. To restore the default time, use the **no** form of this command.

**tcp synwait-time** *seconds*
**no tcp synwait-time** *seconds*

## Syntax Description

| | |
|---|---|
| *seconds* | Time (in seconds) the software waits while attempting to establish a TCP connection. Range is 5 to 30 seconds. |

## Command Default

The default value for the synwait-time is 30 seconds.

## Command Modes

XR Config mode

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read, write |

## Examples

The following example shows how to configure the software to continue attempting to establish a TCP connection for 18 seconds:

```
RP/0/RP0/CPU0:router(config)# tcp synwait-time 18
```

# tcp timestamp

To more accurately measure the round-trip time of a packet, use the **tcp timestamp** command inXR Config mode. To reset the default, use the **no** form of this command.

**tcp timestamp**
**no tcp timestamp**

## Syntax Description

This command has no keywords or arguments.

## Command Default

A TCP time stamp is not used.

## Command Modes

XR Config mode

## Usage Guidelines

Use the **tcp timestamp** command to more accurately measure the round-trip time of a packet. If a time stamp is not used, a TCP sender deduces the round-trip time when an acknowledgment of its packet is received, which is not a very accurate method because the acknowledgment can be delayed, duplicated, or lost. If a time stamp is used, each packet contains a time stamp to identify packets when acknowledgments are received and the round-trip time of that packet.

This feature is most useful in Long Fat Network (LFN) where the bandwidth * delay product is long.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read, write |

## Examples

The following example shows how to enable the timestamp option:

```
RP/0/RP0/CPU0:router(config)# tcp timestamp
```

## Related Commands

| Command | Description |
|---|---|
| tcp selective-ack, on page 570 | Enables the TCP selective acknowledgment feature. |

# tcp window-size

To alter the TCP window size, use the **tcp window-size** command in XR Config mode. To restore the default value, use the **no** form of this command.

**tcp window-size** *bytes*
**no tcp window-size**

## Syntax Description

| | |
|---|---|
| bytes | Window size in bytes. Range is 2048 to 65535 bytes. |

## Command Default

The default value for the window size is 16k.

## Command Modes

XR Config mode

## Usage Guidelines

**Note** Do not use this command unless you clearly understand why you want to change the default value.

## Task ID

| Task ID | Operations |
|---|---|
| transport | read, write |

## Examples

The following example shows how to set the TCP window size to 3000 bytes:

```
RP/0/RP0/CPU0:router(config)# tcp window-size 3000
```

# VRRP Commands

This document describes the Cisco IOS XR software commands used to configure and monitor the Virtual Router Redundancy Protocol (VRRP ).

For detailed information about VRRP concepts, configuration tasks, and examples, refer to the *IP Addresses and Services Configuration Guide for Cisco NCS 6000 Series Routers*.

# accept-mode

To disable the installation of routes for the Virtual Router Redundancy Protocol (VRRP) virtual addresses, use the **accept-mode** command in the VRRP virtual router submode. To enable the installation of routes for the VRRP virtual addresses, use the **no** form of this command.

**accept-mode disable**

**no accept-mode disable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables the accept mode. |

## Command Default

By default, the accept mode is enabled.

## Command Modes

VRRP virtual router configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to disable the installation of routes for the VRRP virtual addresses:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 3 version 2
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# accept-mode disable
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| address (VRRP), on page 580 | Sets the primary virtual IPv4 address for a virtual router. |
| address global, on page 582 | Configures the global virtual IPv6 address for a virtual router. |
| address linklocal, on page 584 | Sets the virtual link-local IPv6 address for a virtual router. |

| Command | Description |
|---|---|
| address secondary, on page 586 | Sets the secondary virtual IPv4 address for a virtual router. |
| message state disable, on page 593 | Disables the task of logging the VRRP state change events. |

# accept-mode (subordinate)

To disable the installation of routes for the Virtual Router Redundancy Protocol (VRRP) virtual addresses, use the **accept-mode** command in the VRRP slave submode. To enable the installation of routes for the VRRP virtual addresses, use the **no** form of this command.

**accept-mode disable**

**no accept-mode disable**

## Syntax Description

| | |
|---|---|
| **disable** | Disables the accept mode. |

## Command Default

By default, the accept mode is enabled.

## Command Modes

VRRP slave submode configuration

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to disable the installation of routes for the VRRP virtual addresses:

```
Router# configure
Router(config)# router vrrp
Router(config-vrrp)# interface tenGigE 0/4/0/4
Router(config-vrrp-if)# address-family ipv4
Router(config-vrrp-address-family)# vrrp 3 slave
Router(config-vrrp-virtual-router)# accept-mode disable
Router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| accept-mode, on page 576 | Disable the installation of routes for the Virtual Router Redundancy Protocol (VRRP) virtual addresses. |

# address-family

To enable address-family mode, use the **address-family** command in interface configuration mode. To terminate address-family mode, use the **no** form of this command.

**address-family** {**ipv4** | **ipv6**}
**no address-family** {**ipv4** | **ipv6**}

## Syntax Description

| | |
|---|---|
| **ipv4** | IPv4 address-family. |
| **ipv6** | IPv6 address-family. |

## Command Default

None.

## Command Modes

Interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

The following example shows how to enable address-family mode:

```
RP/0/RP0/CPU0:router # config
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface hundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
```

## Related Commands

| Command | Description |
|---|---|
| interface (VRRP), on page 591 | Enables VRRP interface configuration mode. |

# address (VRRP)

To configure the primary virtual IPv4 address for a virtual router, use the **address** command in the Virtual Router Redundancy Protocol (VRRP) virtual router submode. To deconfigure the primary virtual IPv4 address for the virtual router, use the **no** form of this command.

**address** *address*

**no address** *address*

## Syntax Description

| | |
|---|---|
| *address* | VRRP IPv4 address. |

## Command Default

None

## Command Modes

VRRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to set the primary virtual IPv4 address for the virtual router:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 3 version 3
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# address 192.168.18.1
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| accept-mode, on page 576 | Disables the installation of routes for the VRRP virtual addresses. |
| address global, on page 582 | Configures the global virtual IPv6 address for a virtual router. |
| address linklocal, on page 584 | Sets the virtual link-local IPv6 address for a virtual router. |

| Command | Description |
| --- | --- |
| address secondary, on page 586 | Sets the secondary virtual IPv4 address for a virtual router. |
| message state disable, on page 593 | Disables the task of logging the VRRP state change events. |

# address global

To configure the global virtual IPv6 address for a virtual router, use the **address global** command in the Virtual Router Redundancy Protocol (VRRP) virtual router submode. To deconfigure the global virtual IPv6 address for a virtual router, use the **no** form of this command.

**address global** *ipv6-address*

**no address global** *ipv6-address*

## Syntax Description

| | |
|---|---|
| *ipv6-address* | Global VRRP IPv6 address. |

## Command Default

None

## Command Modes

VRRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to add a global virtual IPv6 address for the virtual router:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv6
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 3
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# address global 4000::1000
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| address (VRRP), on page 580 | Sets the primary virtual IPv4 address for a virtual router. |
| accept-mode, on page 576 | Disables the installation of routes for the VRRP virtual addresses. |
| address linklocal, on page 584 | Sets the virtual link-local IPv6 address for a virtual router. |

| Command | Description |
| --- | --- |
| address secondary, on page 586 | Sets the secondary virtual IPv4 address for a virtual router. |
| message state disable, on page 593 | Disables the task of logging the VRRP state change events. |

# address linklocal

To either configure the virtual link-local IPv6 address for a virtual router or to specify that the virtual link-local IPv6 address should be enabled and calculated automatically from the virtual router virtual Media Access Control (MAC) address, use the **address linklocal** command in the Virtual Router Redundancy Protocol (VRRP) virtual router submode. To deconfigure the virtual link-local IPv6 address for a virtual router, use the **no** form of this command.

**address linklocal** [{*ipv6-address* | **autoconfig**}]

**no address linklocal** [{*ipv6-address* | **autoconfig**}]

## Syntax Description

| | |
|---|---|
| *ipv6-address* | VRRP IPv6 link-local address. |
| **autoconfig** | Autoconfigures the VRRP IPv6 link-local address. |

## Command Default

None

## Command Modes

VRRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to autoconfigure the VRRP IPv6 link-local address:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router vrrp
RP/0/RP0/CPU0:router(config-vrrp)#interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)#address-family ipv6
RP/0/RP0/CPU0:router(config-vrrp-address-family)#vrrp 3
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#address linklocal autoconfig
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

This example shows how to configure the virtual link-local IPv6 address for the virtual router:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router vrrp
RP/0/RP0/CPU0:router(config-vrrp)#interface HundredGigE 0/4/0/4
```

```
RP/0/RP0/CPU0:router(config-vrrp-if)#address-family ipv6
RP/0/RP0/CPU0:router(config-vrrp-address-family)#vrrp 3
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#address linklocal FE80::260:3EFF:FE11:6770

RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

**Note** The **version** keyword is available only if IPv4 address-family is selected. By default, version is set to 3 for IPv6 address families.

## Related Commands

| Command | Description |
|---|---|
| address (VRRP), on page 580 | Sets the primary virtual IPv4 address for a virtual router. |
| address global, on page 582 | Configures the global virtual IPv6 address for a virtual router. |
| accept-mode, on page 576 | Disables the installation of routes for the VRRP virtual addresses. |
| address secondary, on page 586 | Sets the secondary virtual IPv4 address for a virtual router. |
| message state disable, on page 593 | Disables the task of logging the VRRP state change events. |

# address secondary

To configure the secondary virtual IPv4 address for a virtual router, use the **address secondary** command in the Virtual Router Redundancy Protocol (VRRP) virtual router submode. To deconfigure the secondary virtual IPv4 address for a virtual router, use the **no** form of this command.

**address** *address* **secondary**

**no address** *address* **secondary**

## Syntax Description

| | |
|---|---|
| **secondary** | Sets the secondary VRRP IP address. |
| *address* | VRRP IPv4 address. |

## Command Default

None

## Command Modes

VRRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to set the secondary virtual IPv4 address for the virtual router:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 3 version 2
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# address 192.168.18.1 secondary
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| address (VRRP), on page 580 | Sets the primary virtual IPv4 address for a virtual router. |
| address global, on page 582 | Configures the global virtual IPv6 address for a virtual router. |

| Command | Description |
|---|---|
| address linklocal, on page 584 | Sets the virtual link-local IPv6 address for a virtual router. |
| accept-mode, on page 576 | Disables the installation of routes for the VRRP virtual addresses. |
| message state disable, on page 593 | Disables the task of logging the VRRP state change events. |

# clear vrrp statistics

To reset the Virtual Router Redundancy Protocol (VRRP) statistics (to zero or default value), use the **clear vrrp statistics** command in XR EXEC mode.

**clear vrrp statistics** {**ipv4** | **ipv6**}[**interface** *type* *interface-path-id* [*vrid*]]

## Syntax Description

| | |
|---|---|
| **ipv4** | (Optional) Resets the IPv4 information. |
| **ipv6** | (Optional) Resets the IPv6 information. |
| **interface** *type* | (Optional) Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | (Optional) Either a physical interface instance or a virtual interface instance as follows:<br>• Physical interface instance. Naming notation is *rack/slot/module/port* and a slash between values is required as part of the notation.<br>  • *rack*: Chassis number of the rack.<br>  • *slot*: Physical slot number of the modular services card or line card.<br>  • *module*: Module number. A physical layer interface module (PLIM) is always 0.<br>  • *port*: Physical port number of the interface.<br>**Note** In references to a Management Ethernet interface located on a route processor card, the physical slot number is alphanumeric (RP0 or RP1) and the module is CPU0. Example: interface mgmtEth 0/RP1/CPU0/0.<br>• Virtual interface instance. Number range varies depending on interface type.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **vrid** | (Optional) Virtual router identifier, which is the number identifying the virtual router for which status is displayed. |

## Command Default

No default behavior or values

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If no **interface** is specified, the statistics for all virtual routers on all interfaces are cleared.

If no value for *vrid* is specified, the statistics for all virtual routers on the specified interface are cleared.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

The following example shows how to clear vrrp statistics:

```
RP/0/RP0/CPU0:router# clear vrrp statistics
```

## Related Commands

| Command | Description |
|---|---|
| show vrrp | Displays a brief or detailed status of one or all Virtual Router Redundancy Protocol (VRRP) virtual routers. |

# delay (VRRP)

To configure the activation delay for a VRRP router, use the **delay** command in interface configuration mode. To delete the activation delay, use the **no** form of this command.

**delay minimum** *value* **reload** *value*
**no delay**

## Syntax Description

| | |
|---|---|
| **minimum** *value* | Sets the minimum delay in seconds for every interface up event. Range is 0 to 10000. |
| **reload** *value* | Sets the reload delay in seconds for first interface up event. Range is 0 to 10000. |

## Command Default

**minimum** *value:* 1

**reload** *value:* 5

## Command Modes

VRRP interface configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **vrrp delay** command delays the start of the VRRP finite state machine (FSM) on an interface up event to ensure that the interface is ready to pass traffic. This ensures that there are no mistaken state changes due to loss of hello packets. The minimum delay is applied on all interface up events and the reload delay is applied on the first interface up event.

The values of zero must be explicitly configured to turn this feature off.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

The following example shows how to configure a minimum delay of 10 seconds with a reload delay of 100 seconds:

```
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface /CPU0/0
RP/0/RP0/CPU0:router(config-vrrp-if)# delay minimum 10 reload 100
```

## Related Commands

| Command | Description |
|---|---|
| show vrrp | Displays a brief or detailed status of one or all Virtual Router Redundancy Protocol (VRRP) virtual routers. |

# interface (VRRP)

To enable VRRP interface configuration mode, use the **interface (VRRP)** command in VRRP configuration mode. To terminate VRRP interface configuration mode, 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.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |

## Command Default

VRRP is disabled.

## Command Modes

VRRP configuration

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **interface (VRRP)** command to enter VRRP interface configuration mode.

You must configure all VRRP configuration commands in VRRP interface configuration mode.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

The following example shows how to configure VRRP and a virtual router 1 on 10-Gigabit Ethernet interface 0/3/0/0:

```
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface TenGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-vrrp-if)# vrrp 1 ipv4 192.168.18.1
```

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 3 version 2
```

```
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| router vrrp, on page 594 | Configures a VRRP redundancy process. |

# message state disable

To disable the task of logging the Virtual Router Redundancy Protocol (VRRP) state change events via syslog, use the **message state disable** command in the VRRP virtual router submode. To re-enable the task of logging the VRRP state change events , use the **no** form of this command.

**message state disable**

**no message state disable**

## Syntax Description

This command has no keywords or arguments.

## Command Default

By default, the task of logging the VRRP state change events is enabled.

## Command Modes

VRRP global

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to disable the logging of VRRP state change events:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#router vrrp
RP/0/RP0/CPU0:router(config-vrrp)#message state disable
RP/0/RP0/CPU0:router(config-vrrp)#
```

## Related Commands

| Command | Description |
|---|---|
| address (VRRP), on page 580 | Sets the primary virtual IPv4 address for a virtual router. |
| address global, on page 582 | Configures the global virtual IPv6 address for a virtual router. |
| accept-mode, on page 576 | Disables the installation of routes for the VRRP virtual addresses. |
| address secondary, on page 586 | Sets the secondary virtual IPv4 address for a virtual router. |
| address linklocal, on page 584 | Sets the virtual link-local IPv6 address for a virtual router. |

# router vrrp

To configure Virtual Router Redundancy Protocol (VRRP), use the **router vrrp** command in XR Config mode. To remove the VRRP configuration, use the **no** form of this command.

**router vrrp**
**no router vrrp**

## Command Default

This command has no keywords or arguments.

VRRP is disabled.

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use the **router vrrp** command to enter VRRP configuration mode.

You must configure all VRRP configuration commands in VRRP interface configuration mode.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

The following example shows how to configure a VRRP with virtual router 1 on an interface:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 3 version 2
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| interface (VRRP), on page 591 | Enables VRRP interface configuration mode. |

# session name(vrrp)

To configure a VRRP session name, use the **session name** command in the VRRP virtual router submode. To deconfigure a VRRP session name, use the **no** form of this command.

**name** *name*
**no name** *name*

## Syntax Description

| | |
|---|---|
| *name* | MGO session name |

## Command Default

None

## Command Modes

VRRP virtual router configuration

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read |

### Example

This example shows how to configure a VRRP session name.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface tenGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-ipv4)# vrrp 1
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# name s1
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| accept-mode, on page 576 | Disable the installation of routes for the Virtual Router Redundancy Protocol (VRRP) virtual addresses. |

# show vrrp

To display a brief or detailed status of one or all Virtual Router Redundancy Protocol (VRRP) virtual routers, use the **show vrrp** command in XR EXEC mode.

**show vrrp** [{**ipv4** | **ipv6**}] [**interface** *type interface-path-id* ] [{**brief** | **detail** | **statistics** [**all**]}]

## Syntax Description

| | |
|---|---|
| **ipv4** | (Optional) Displays the IPv4 information. |
| **ipv6** | (Optional) Displays the IPv6 information. |
| **interface** | (Optional) Displays the status of the virtual router interface. |
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the show interfaces command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| **brief** | (Optional) Provides a summary view of the virtual router information. |
| **detail** | (Optional) Displays detailed running state information. |
| **statistics** | (Optional) Displays total statistics. |
| **all** | (Optional) Displays statistics for each virtual router. |

## Command Default

None

## Command Modes

XR EXEC mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

If no interface is specified, all virtual routers on all interfaces are displayed. If no vrid is specified, all vrids on the given interface are displayed.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read |

## Examples

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

```
Router# show vrrp

                     A indicates IP address owner
                     | P indicates configured to preempt
                     | |
Interface   vrID Prio A P State    Master addr     VRouter addr
Te0/3/0/0      1  100   P Init     unknown         192.168.18.10
Te0/3/0/2      7  100   P Init     unknown         192.168.19.1
```

This table describes the significant fields shown in the display.

***Table 86: show vrrp Command Field Descriptions***

| Field | Description |
|---|---|
| Interface | Interface of the virtual router. |
| vrID | ID of the virtual router. |
| Prio | Priority of the virtual router. |
| A | Indicates whether the VRRP router is the IP address owner. |
| P | Indicates whether the VRRP router is configured to preempt (default). |
| State | State of the virtual router. |
| Master addr | IP address of the IP address owner router. |
| VRouter addr | Virtual router IP address of the virtual router. |

The following sample output is from the **show vrrp** command with the **detail** keyword:

```
Router# show vrrp detail
HundredGigE0/4/0/0 - IPv4 vrID 1
  State is Master, IP address owner
    2 state changes, last state change 00:00:59
```

```
  Virtual IP address is 192.168.10.1
    Secondary Virtual IP address is 192.168.10.2
    Secondary Virtual IP address is 192.168.11.1
  Virtual MAC address is 0000.5E00.0101
  Master router is local
  Advertise time 1 secs
    Master Down Timer 3.609 (3 x 1 + 156/256)
  Minimum delay 1 sec, reload delay 5 sec
  Current priority 100
    Configured priority 110, may preempt
      Minimum delay 0 secs
  Authentication enabled, string "myauth"
  BFD enabled: state Up, interval 15ms multiplier 3 remote IP 192.168.10.3
    Tracked items:
                                      Priority
    Interface                State     Decrement
    HundredGigE0/5/0/1               Down            10

HundredGigE0/4/0/0 - IPv4 vrID 2
  State is Backup
    3 state changes, last state change 00:01:58
  Virtual IP address is 192.168.10.2
  Virtual MAC address is 0000.5E00.0102
  Master router is IP address owner (192.168.11.1), priority 200
  Advertise time 1.500 secs (forced)
    Master Down Timer 5.109 (3 x 1 + 156/256)
  Minimum delay 1 sec, reload delay 5 sec
  Current priority 100
    Configured priority 100, may preempt
      Minimum delay 20 secs


Bundle-Ether1 - IPv4 vrID 5
  State is Init
    0 state changes, last state change never
  Virtual IP address is unknown
  Virtual MAC address is 0000.5E00.0100
  Master router is unknown
  Advertise time 1 secs
    Master Down Timer 3.500 (3 x 1 + 128/256)
  Minimum delay 1 sec, reload delay 5 sec
  Current priority 128
    Configured priority 128

HundredGigE0/4/0/0 - IPv6 vrID 1
  State is Master
    2 state changes, last state change 00:10:01
  Virtual Linklocal address is FE80::100
    Global Virtual IPv6 address is 4000::100
    Global Virtual IPv6 address is 5000::100
  Virtual MAC address is 0000.5E00.0201
  Master router is local
  Advertise time 1 secs
    Master Down Timer 3.609 (3 x 1 + 156/256)
  Minimum delay 1 sec, reload delay 5 sec
  Current priority 100
    Configured priority 100, may preempt
      Minimum delay 0 secs
```

This table describes the significant fields shown in the displays.

***Table 87: show vrrp detail Command Field Descriptions***

| Field | Description |
|---|---|
| HundredGigE 0/3/0/0 - vrID 1 | Interface type and number, and VRRP group number. |
| State is | Role this interface plays within VRRP (IP address owner router or backup router). |
| Virtual IP address is | Virtual IP address for this virtual router. |
| Virtual MAC address is | Virtual MAC address for this virtual router. |
| Master router is | Location of the IP address owner router. |
| Advertise time | Interval (in seconds) at which the router sends VRRP advertisements when it is the IP address owner virtual router. This value is configured with the **vrrp timer** command. |
| Master Down Timer | Time the backup router waits for the IP address owner router advertisements before assuming the role of IP address owner router. |
| Minimum delay | Time that the state machine start-up is delayed when an interface comes up, giving the network time to settle. The minimum delay is the delay that is applied after any subsequent interface up event (if the interface flaps) and the reload delay is the delay applied after the first interface up event. |
| Current priority | Priority of the virtual router. |
| Configured priority | Priority configured on the virtual router. |
| may preempt | Indication of whether preemption is enabled or disabled. |
| minimum delay | Delay time before preemption (default) occurs. |
| Tracked items | Section indicating the items being tracked by the VRRP router. |
| Interface | Interface being tracked. |
| State | State of the tracked interface. |
| Priority Decrement | Priority to decrement from the VRRP priority when the interface is down. |

The following sample output is from the **show vrrp** command with the **interface** and **detail** keywords for Ethernet interface 0/3/0/0:

```
Router# show vrrp interface gigabitEthernet 0/3/0/0

                     A indicates IP address owner
                     | P indicates configured to preempt
                     | |
Interface   vrID Prio A P State    Master addr     VRouter addr
```

```
Te0/3/0/0       1  100   P Init     unknown         192.168.10.20
Te0/3/0/2       7  100   P Init     unknown         192.168.20.0
```

***Table 88: show vrrp interface Command Field Descriptions***

| Field | Description |
|---|---|
| Interface | Interface of the virtual router. |
| vrID | ID of the virtual router. |
| Prio | Priority of the virtual router. |
| A | Indicates whether the VRRP router is the IP address owner. |
| P | Indicates whether the VRRP router is configured to preempt (default). |
| State | State of the virtual router. |
| Master addr | IP address of the IP address owner router. |
| VRouter addr | Virtual router IP address of the virtual router. |

# vrrp slave follow

To instruct the subordinate group to inherit its state from a specified group, use the **vrrp slave follow** command in VRRP slave submode.

**follow** *mgo-session-name*

## Syntax Description

| | |
|---|---|
| *mgo-session-name* | Name of the MGO session from which the subordinate group will inherit the state. |

## Command Default

None

## Command Modes

VRRP slave submode configuration

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to instruct the subordinate group to inherit its state from a specified group.

```
Router# configure
Router(config)# router vrrp
Router(config-vrrp)# interface tenGigE 0/4/0/4
Router(config-vrrp-if)# address-family ipv4
Router(config-vrrp-address-family)# vrrp 2 slave
Router(config-vrrp-slave)# follow m1
```

**Note** Before configuring a subordinate group to inherit its state from a specified group, the group must be configured with the **session name** command on another vrrp group.

## Related Commands

| Command | Description |
|---|---|
| accept-mode, on page 576 | Disable the installation of routes for the Virtual Router Redundancy Protocol (VRRP) virtual addresses. |

# subordinate primary virtual IPv4 address(vrrp)

To configure the primary virtual IPv4 address for the subordinate group, use the **subordinate primary virtual IPv4 address** command in the VRRP slave submode.

**address** *ip-address*

## Syntax Description

| | |
|---|---|
| *ip-address* | IP address of the Hot Standby router interface. |

## Command Default

None

## Command Modes

VRRP slave submode configuration

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to configure the primary virtual IPv4 address for the subordinate group.

```
Router# configure
Router(config)# router vrrp
Router(config-vrrp)# interface tenGigE 0/4/0/4
Router(config-vrrp-if)# address-family ipv4
Router(config-vrrp-address-family)# vrrp 2 slave
Router(config-vrrp-slave)# address 192.168.10.4
```

## Related Commands

| Command | Description |
|---|---|
| accept-mode, on page 576 | Disable the installation of routes for the Virtual Router Redundancy Protocol (VRRP) virtual addresses. |

# subordinate secondary virtual IPv4 address(vrrp)

To configure the secondary virtual IPv4 address for the subordinate group, use the **subordinate secondary virtual IPv4 address** command in the VRRP slave submode.

**address** *ip-address* **secondary**

## Syntax Description

| | |
|---|---|
| *ip-address* | IP address of the Hot Standby router interface. |
| **secondary** | Sets the secondary hot standby IP address. |

## Command Default

None

## Command Modes

VRRP slave submode configuration

## Usage Guidelines

Before configuring secondary virtual IPv4 address, the primary virtual IPv4 address for the subordinate group must be configured.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to configure the secondary virtual IPv4 address for the subordinate group.

```
Router# configure
Router(config)# router vrrp
Router(config-vrrp)# interface tenGigE 0/4/0/4
Router(config-vrrp-if)# address-family ipv4
Router(config-vrrp-address-family)# vrrp 2 slave
Router(config-vrrp-slave)# address 192.168.10.4 secondary
```

## Related Commands

| Command | Description |
|---|---|
| accept-mode, on page 576 | Disable the installation of routes for the Virtual Router Redundancy Protocol (VRRP) virtual addresses. |

# snmp-server traps vrrp events

To enable the Simple Network Management Protocol (SNMP) server notifications (traps) available for VRRP, use the **snmp-server traps vrrp events command** in XR Config mode. To disable all available VRRP SNMP notifications, use the **no** form of this command.

**snmp-server traps vrrp events**
**no snmp-server traps vrrp events**

## Syntax Description

| | |
|---|---|
| **events** | Specifies all VRRP SNMP server traps. |

## Command Default

None

## Command Modes

XR Config mode

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| snmp | read, write |

## Examples

The following example shows how to enable snmpserver notifications for VRRP:

```
RP/0/RP0/CPU0:routerrouter(config)# snmp-server traps vrrp events
```

# track object(vrrp)

To enable tracking of a named object with the specified decrement, use the **track object** command in VRRP virtual router submode. To remove the tracking, use the **no** form of this command.

**track object** *name*[*priority-decrement*]
**no track object** *name*[*priority-decrement*]

## Syntax Description

| | |
|---|---|
| **object** *name* | Object tracking. Name of the object to be tracked. |
| *priority-decrement* | (Optional) Amount by which the VRRP priority for the router is decremented when the interface goes down (or comes back up). Range is 1 to 255. |

## Command Default

The default priority-decrement is 10.

## Command Modes

VRRP virtual router configuration

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

This example shows how to configure object tracking under the VRRP virtual router submode.

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface tenGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-ipv4)# vrrp 1
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# track object t1 2
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
|---|---|
| accept-mode, on page 576 | Disable the installation of routes for the Virtual Router Redundancy Protocol (VRRP) virtual addresses. |

# vrrp

To enable Virtual Router Redundancy Protocol (VRRP) virtual router mode, use the **vrrp** command in address-family mode. To terminate VRRP virtual router mode, use the **no** form of this command.

**vrrp** *vrid* **version** *version-no*
**novrrp** *vrid* **version** *version-no*

## Syntax Description

| | |
|---|---|
| *vrid* | (Optional) Virtual router identifier, which is the number identifying the virtual router for which status is displayed. The virtual router identifier is configured with the vrrp ipv4 command. Range is 1 to 255. |
| **version** *version-no* | The VRRP version number. Range is 2-3. |

**Note** The **version** keyword is available only for the ipv4 address family. By default, version is set to 3 for IPv6 address families.

## Command Default

None.

## Command Modes

address-family

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operation |
|---|---|
| vrrp | read, write |

### Example

The following example shows how to enable VRRP virtual router mode:

```
RP/0/RP0/CPU0:router# config
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/4/0/4
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 3 version 2
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)#
```

## Related Commands

| Command | Description |
| --- | --- |
| interface (VRRP), on page 591 | Enables VRRP interface configuration mode. |

# vrrp preempt

VRRP preempt is enabled by default. This means, a VRRP router with higher priority than the current IP address owner router will take over as new IP address owner router. To disable this feature, use the **preempt disable** command. To delay preemption, so that the higher priority router waits for a period of time before taking over, use the **preempt delay** command. To restore the default behavior (preempt enabled with no delay), use the **no** form of the command.

**preempt** {**delay** *seconds* | **disable**}
**no preempt** {**delay** *seconds* | **disable**}

## Syntax Description

| | |
|---|---|
| **delay** *seconds* | Specifies the number of seconds the router delays before issuing an advertisement claiming virtual IP address ownership to be the IP address owner router. Range is 1 to 3600 seconds (1 hour). |
| **disable** | Disables preemption |

## Command Default

VRRP preempt is enabled.

*seconds* : 0 (no delay)

## Command Modes

VRRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

, can configure a delay, which causes the VRRP router to wait the specified number of seconds before issuing an advertisement claiming virtual IP address ownership to be the IP address owner router.

**Note** The router that is the virtual IP address owner preempts, regardless of the setting of this command.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

The following example shows how to configure the router to preempt the current IP address owner router when its priority of 200 is higher than that of the current IP address owner router. If the router preempts the current IP address owner router, it waits 15 seconds before issuing an advertisement claiming that it is the new IP address owner router.

```
Router(config)# router vrrp
Router(config-vrrp)# interface HundredGigE 0/3/0/0
Router(config-vrrp-if)# address-family ipv4
```

```
Router(config-vrrp-address-family)# vrrp 1 version 3
Router(config-vrrp-virtual-router)# preempt delay 15
Router(config-vrrp-virtual-router)# priority 200
```

## Related Commands

| Command | Description |
|---|---|
| vrrp priority, on page 610 | Sets the priority of the virtual router. |

# vrrp priority

To set the priority of the virtual router, use the **priority** command in VRRP virtual router submode. To remove the priority of the virtual router, use the **no** form of this command.

**priority** *priority*
**nopriority** *priority*

## Syntax Description

| | |
|---|---|
| *priority* | Priority of the virtual router. Range is 1 to 254. |

## Command Default

*priority* : 100

## Command Modes

VRRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

Use this command to control which router becomes the IP address owner router. This command is ignored while the router is the virtual IP address owner.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

The following example shows how to configure the router with a priority of 254:

```
Router(config)# router vrrp
Router(config-vrrp)# interface HundredGigE 0/3/0/0
Router(config-vrrp-if)# address-family ipv4
Router(config-vrrp-address-family)# vrrp 1 version 3
Router(config-vrrp-virtual router)# priority 254
```

## Related Commands

| Command | Description |
|---|---|
| vrrp preempt, on page 608 | |

# vrrp text-authentication

To configure the simple text authentication used for Virtual Router Redundancy Protocol (VRRP) packets received from other routers running VRRP, use the **text-authentication** command in VRRP virtual router submode. To disable VRRP authentication, use the **no** form of this command.

**text-authentication** *string*
**no text-authentication** [*string*]

## Syntax Description

| | |
|---|---|
| *string* | Authentication string (up to eight alphanumeric characters) used to validate incoming VRRP packets. |

## Command Default

No authentication of VRRP messages occurs.

## Command Modes

VRRP virtual router

## Usage Guidelines

When a VRRP packet arrives from another router in the VRRP group, its authentication string is compared to the string configured on the local system. If the strings match, the message is accepted. If they do not match, the packet is discarded.

All routers within the group must be configured with the same authentication string.

**Note** Plain text authentication is not meant to be used for security. It simply provides a way to prevent a misconfigured router from participating in VRRP.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

The following example shows how to configure an authentication string of x30dn78k:

```
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 1 version 2
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# text-authentication x30dn78k
```

**Note** Text authentication is only valid for VRRP version 2 routers.

# vrrp timer

To configure the interval between successive advertisements by the IP address owner router in a Virtual Router Redundancy Protocol (VRRP) virtual router, use the **timer** command in VRRP virtual router submode. To restore the default value, use the **no** form of this command.

**timer** [**msec**] *interval* [**force**]
**no timer** [**msec**] *interval* [**force**]

## Syntax Description

| | |
|---|---|
| **msec** | (Optional) Changes the unit of the advertisement time from seconds to milliseconds. Without this keyword, the advertisement interval is in seconds. Range is 20 to 3000 milliseconds. |
| *interval* | Time interval between successive advertisements by the IP address owner router. The unit of the interval is in seconds, unless the **msec** keyword is specified. Range is 1 to 255 seconds. |
| **force** | (Optional) Forces the configured value to be used. This keyword is required if milliseconds is specified. |

## Command Default

*interval*:1 second

## Command Modes

VRRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

No specific guidelines impact the use of this command.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

The following example shows how to configure the IP address owner router to send advertisements every 4 seconds:

```
Router(config)# router vrrp
Router(config-vrrp)# interface HundredGigE 0/3/0/0
Router(config-vrrp-if)# address-family ipv4
Router(config-vrrp-address-family)# vrrp 1 version 3
Router(config-vrrp-virtual-router)# timer 4
```

# vrrp track interface

To configure the Virtual Router Redundancy Protocol (VRRP) to track an interface, use the **track interface** command in VRRP virtual router submode. To disable the tracking, use the **no** form of this command.

**track interface** *type interface-path-id* [*priority-decrement*]
**no track interface** *type interface-path-id* [*priority-decrement*]

## Syntax Description

| | |
|---|---|
| *vrid* | Virtual router identifier, which is the number identifying the virtual router to which tracking applies. |
| *type* | Interface type. For more information, use the question mark (?) online help function. |
| *interface-path-id* | Physical interface or virtual interface.<br>**Note** Use the **show interfaces** command to see a list of all interfaces currently configured on the router.<br>For more information about the syntax for the router, use the question mark (?) online help function. |
| *priority-decrement* | (Optional) Amount by which the priority for the router is decremented (or incremented) when the tracked interface goes down (or comes back up). Decrements can be set to any value between 1 and 254. Default value is 10. |

## Command Default

The default decrement value is 10. Range is 1 to 254.

## Command Modes

VRRP virtual router

## Command History

| Release | Modification |
|---|---|
| Release 5.0.0 | This command was introduced. |

## Usage Guidelines

The **vrrp track interface** command ties the priority of the router to the availability of its interfaces. It is useful for tracking interfaces that are not configured for VRRP. Only IP interfaces are tracked. A tracked interface is up if IP on that interface is up. Otherwise, the tracked interface is down.

You can configure VRRP to track an interface that can alter the priority level of a virtual router for a VRRP virtual router. When the IP protocol state of an interface goes down or the interface has been removed from the router, the priority of the backup virtual router is decremented by the value specified in the *priority-decrement* argument. When the IP protocol state on the interface returns to the up state, the priority is restored.

## Task ID

| Task ID | Operations |
|---|---|
| vrrp | read, write |

## Examples

In the following example, 10-Gigabit Ethernet interface 0/3/0/0 tracks interface 0/3/0/3 and 0/3/0/2. If one or both of these two interfaces go down, the priority of the router decreases by 10 (default priority decrement) for each interface. The default priority decrement is changed using the *priority-decrement* argument. In this example, because the default priority of the virtual router is 100, the priority becomes 90 when one of the tracked interfaces goes down and the priority becomes 80 when both go down. See the **priority** command for details on setting the priority of the virtual router.

```
RP/0/RP0/CPU0:router(config)# router vrrp
RP/0/RP0/CPU0:router(config-vrrp)# interface HundredGigE 0/3/0/0
RP/0/RP0/CPU0:router(config-vrrp-if)# address-family ipv4
RP/0/RP0/CPU0:router(config-vrrp-address-family)# vrrp 1 version 3
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# track interface HundredGigE 0/3/0/3
RP/0/RP0/CPU0:router(config-vrrp-virtual-router)# track interface HundredGigE 0/3/0/2
```

## Related Commands

| Command | Description |
|---|---|
| vrrp priority, on page 610 | Sets the priority of the virtual router. |

# INDEX

## G



## H



## I



## L



## M



## N



## P



## R



## S

## T

## V