IPv6 Command Reference, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)

First Published: January 11, 2013

Last Modified: January 22, 2013

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

© 2013 Cisco Systems, Inc. All rights reserved.

CONTENTS

ipv6-a1

allow

To limit the number of multicast router advertisements (RAs) per device per throttle period in an RA throttler policy, use the **allow** command in IPv6 RA throttle policy configuration mode. To reset the command to its defaults, use the **no** form of this command.

allow {**at-least** | {*al-value*| **no-limit**}} | {**at-most** | {*am-value*| **no-limit**}} | {**inherited**}

Syntax Description

at-least	The minimum number of multicast RAs accepted from the device before throttling occurs.
al-value	At-least value. • An integer from 0 through 32.
no-limit	No RA throttling will occur.
at-most	The maximum number of multicast RAs accepted from the device before throttling occurs.
am-value	At-most value. • An integer from 0 through 256.
inherited	The setting between target policies is inherited, or coalesced.

Command Default

The **at-least** value is 1.

The **at-most** value is 1.

Command Modes

IPv6 RA throttle policy configuration mode (config-nd-ra-throttle)

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

The **allow at-least** and **allow at-most** command settings applied at the VLAN level provide the defaults for all devices on the VLAN. If the device that issued the RA has not yet sent the number of RAs configured by the **allow at-least** command setting, then the RA is multicast to all hosts. If the device that issued the RA has sent the number of RAs configured by the **allow at-most** command setting, then the RA is throttled; that is, the RA is multicast to all wired hosts and to wireless hosts with pending router solicitations (RSs).

If your deployment has the same setting for the **allow at-least** and **allow at-most** values for all devices on all ports, then you only need to apply the policy on the relevant VLAN or VLANs. If some of the wired ports in the deployment are connection wireless access points, then a policy with only the medium type configured needs to be applied on those specific ports.

Examples

```
Device(config)# ipv6 nd ra-throttle policy policy1
Device(config-nd-ra-throttle)# allow at-least 2 at-most 2
```

clear bgp ipv6

To reset IPv6 Border Gateway Protocol (BGP) sessions, use the **clear bgp ipv6**command in privileged EXEC mode.

[1]

Syntax Description

unicast	Specifies IPv6 unicast address prefixes.
multicast	Specifies IPv6 multicast address prefixes.
*	Resets all current BGP sessions.
autonomous-system-number	Resets BGP sessions for BGP neighbors within the specified autonomous system.
ip-address	Resets the TCP connection to the specified IPv4 BGP neighbor and removes all routes learned from the connection from the BGP table.
ipv6-address	Resets the TCP connection to the specified IPv6 BGP neighbor and removes all routes learned from the connection from the BGP table. This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.
peer-group-name	Resets the TCP connection to the specified IPv6 BGP neighbor and removes all routes learned from the connection from the BGP table.
soft	(Optional) Soft reset. Does not reset the session.
in **out**	(Optional) Triggers inbound or outbound soft reconfiguration. If the **in** or **out** option is not specified, both inbound and outbound soft resets are triggered.

Command Default

No reset is initiated.

Command Modes

Privileged EXEC

[1]

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.3(2)T	The **unicast** keyword was added to Cisco IOS Release 12.3(2)T.
12.0(26)S	The **unicast** and **multicast** keywords were added to Cisco IOS Release 12.0(26)S.
12.3(4)T	The **multicast**keyword was added to Cisco IOS Release 12.3(4)T.
12.2(25)S	The **multicast**keyword was added to Cisco IOS Release 12.2(25)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Series Routers.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **clear bgp ipv6**command is similar to the **clear ip bgp**command, except that it is IPv6-specific.

Use of the **clear bgp ipv6** command allows a reset of the neighbor sessions with varying degrees of severity depending on the specified keywords and arguments.

Use the **clear bgp ipv6 unicast** command to drop neighbor sessions with IPv6 unicast address prefixes.

The **unicast** keyword is available in Cisco IOS Release 12.3(2)T and later releases. It is not available in releases prior to 12.3(2)T. Use of the **unicast** keyword is mandatory starting with Cisco IOS Release 12.3(2)T.

The **multicast**keyword is available in Cisco IOS Release 12.0(26)S and later releases. It is not available in releases prior to 12.0(26)S. Use of either the **unicast** or **multicast** keyword is mandatory starting with Cisco IOS Release 12.0(26)S.

Use the **clear bgp ipv6 ***command to drop all neighbor sessions. The Cisco IOS software will then reset the neighbor connections. Use this form of the command in the following situations:

- BGP timer specification change
- BGP administrative distance changes

Use t he **clear bgp ipv6 soft out** or the **clear bgp ipv6 unicast soft out**command to drop only the outbound neighbor connections. Inbound neighbor sessions will not be reset. Use this form of the command in the following situations:

- BGP-related access lists change or get additions
- BGP-related weights change
- BGP-related distribution lists change
- BGP-related route maps change

Use the **clear bgp ipv6 soft in**or the **clear bgp ipv6 unicast soft in**command to drop only the inbound neighbor connections. Outbound neighbor sessions will not be reset. To reset inbound routing table updates dynamically for a neighbor, you must configure the neighbor to support the router refresh capability. To determine whether a BGP neighbor supports this capability, use the **show bgp ipv6 neighbors** or the **show bgp ipv6 unicast neighbors**command. If a neighbor supports the route refresh capability, the following message is displayed:

```
Received route refresh capability from peer.
```

If all BGP networking devices support the route refresh capability, use the **clear bgp ipv6** {*****| ip-*address*| *ipv6-address*| *peer-group-name*} **in** or the **clear bgp ipv6 unicast**{*****| ip-*address*| *ipv6-address*| *peer-group-name*} **in**command. Use of the **soft** keyword is not required when the route refresh capability is supported by all BGP networking devices, because the software automatically performs a soft reset.

Use this form of the command in the following situations:

- BGP-related access lists change or get additions
- BGP-related weights change
- BGP-related distribution lists change
- BGP-related route maps change

Examples

The following example clears the inbound session with the neighbor 7000::2 without the outbound session being reset:

```
Router# clear bgp ipv6 unicast 7000::2 soft in
```

The following example uses the **unicast** keyword and clears the inbound session with the neighbor 7000::2 without the outbound session being reset:

```
Router# clear bgp ipv6 unicast 7000::2 soft in
```

The following example clears the outbound session with the peer group named marketing without the inbound session being reset:

```
Router# clear bgp ipv6 unicast marketing soft out
```

The following example uses the **unicast** keyword and clears the outbound session with the peer group named peer-group marketing without the inbound session being reset:

```
Router# clear bgp ipv6 unicast peer-group marketing soft out
```

Related Commands

Command	Description
show bgp ipv6	Displays entries in the IPv6 BGP routing table.

clear ipv6 mtu

To clear the maximum transmission unit (MTU) cache of messages, use the **clear ipv6 mtu**command in privileged EXEC mode.

clear ipv6 mtu

Syntax Description

This command has no arguments or keywords.

Command Default

Messages are not cleared from the MTU cache.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 2.6	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

If a router is flooded with ICMPv6 toobig messages, the router is forced to create an unlimited number of entries in the MTU cache until all available memory is consumed. Use the **clear ipv6 mtu** command to clear messages from the MTU cache.

Examples

The following example clears the MTU cache of messages:

```
Router# clear ipv6 mtu
```

Related Commands

Command	Description
ipv6 flowset	Configures flow-label marking in 1280-byte or larger packets sent by the router.

default-metric (OSPFv3)

To set default metric values for IPv4 and IPv6 routes redistributed into the Open Shortest Path First version 3 (OSPF) routing protocol, use the **default-metric** command in OSPFv3 router configuration mode, IPv6 address family configuration mode, or IPv4 address family configuration mode. To return to the default state, use the **no** form of this command.

default-metric *metric-value*

no default-metric *metric-value*

Syntax Description

metric-value	Default metric value appropriate for the specified routing protocol. The range is from 1 to 4294967295.

Command Default

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

Command Modes

OSPFv3 router configuration mode (config-router)

IPv6 address family configuration (config-router-af)

IPv4 address family configuration (config-router-af)

Command History

Release	Modification
12.2(15)T	This command was introduced.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
15.1(3)S	This command was modified. The feature can be enabled in an IPv4 or IPv6 OSPFv3 process.
Cisco IOS XE Release 3.4S	This command was modified. The feature can be enabled in an IPv4 or IPv6 OSPFv3 process.
15.2(1)T	This command was modified. The feature can be enabled in an IPv4 or IPv6 OSPFv3 process.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **default-metric** command is used in conjunction with the **redistribute** router configuration command to cause the current routing protocol to use the same metric value for all redistributed routes. A default metric helps solve the problem of redistributing routes with incompatible metrics. Whenever metrics do not convert, using a default metric provides a reasonable substitute and enables the redistribution to proceed.

You can gain finer control over the metrics of redistributed routes by using the options for the **redistribute** command.

Examples

The following example shows how to enter IPv6 AF and configure OSPFv3 routing protocol redistributing routes from the OSPFv3 process named process1. All the redistributed routes are advertised with a metric of 10.

```
router ospfv3 100
 address-family ipv6 unicast
 default-metric 10
 redistribute ospfv3 process1
```

The following example shows an OSPFv3 routing protocol redistributing routes from the OSPFv3 process named process1. All the redistributed routes are advertised with a metric of 10.

```
ipv6 router ospf 100
 default-metric 10
 redistribute ospfv3 process1
```

Related Commands

Command	Description
redistribute (OSPFv3)	Redistributes IPv6 routes from one routing domain into another routing domain.
router ospfv3	Enables OSPFv3 router configuration mode for the IPv4 or IPv6 address family.

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.

deny *protocol* {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] **[undetermined-transport]**

no deny *protocol* {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] **[undetermined-transport]**

Internet Control Message Protocol

deny icmp {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] [*icmp-type* [*icmp-code*]| *icmp-message*] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

Transmission Control Protocol

deny tcp {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] **[ack]** [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] **[established] [fin]** [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] [**neq** {*port*| *protocol*}] **[psh]** [**range** {*port*| *protocol*}] **[routing]** [**routing-type** *routing-number*] **[rst]** [**sequence** *value*] **[syn]** [**time-range** *name*] **[urg]**

User Datagram Protocol

deny udp {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] [**neq** {*port*| *protocol*}] [**range** {*port*| *protocol*}] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

Syntax Description

protocol	Name or number of an Internet protocol. It can be one of the keywords **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **sctp**, **tcp**, **udp**, or **hbh**, 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. 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 *source-ipv6-address*	The source IPv6 host address about which to set deny conditions. 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-number*]	(Optional) Specifies an 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). If the operator is positioned after the *source-ipv6-prefix*/*prefix-length* argument, it must match the source port. If the operator is positioned after the *destination-ipv6*/*prefix-length* argument, it must match the destination port. The **range** operator requires two port numbers. All other operators require one port number. The optional *port-number* argument is a decimal number or the name of a TCP or UDP port. A port number is a number from 0 to 65535. 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*	The destination IPv6 network or class of networks about which to set deny conditions. 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*	The destination IPv6 host address about which to set deny conditions. 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.

auth	Allows matching traffic against the presence of the authentication header in combination with any protocol.
dest-option-type	(Optional) Matches IPv6 packets against the hop-by-hop option extension header within each IPv6 packet header.
doh-number	(Optional) Integer in the range from 0 to 255 representing an IPv6 destination option extension header.
doh-type	(Optional) Destination option header types. The possible destination option header type and its corresponding *doh-number* value are home-address—201.
dscp *value*	(Optional) Matches a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63.
flow-label *value*	(Optional) Matches a flow label value against the flow label value in the Flow Label field of each IPv6 packet header. The acceptable range is from 0 to 1048575.
fragments	(Optional) Matches non-initial fragmented packets where the fragment extension header contains a non-zero fragment offset. The **fragments** keyword is an option only if the *operator* [*port-number*] arguments are not specified.
hbh	(Optional) Specifies a hop-by-hop options header.
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.) 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.
mobility	(Optional) Extension header type. Allows matching of any IPv6 packet including a mobility header, regardless of the value of the mobility-header-type field within that header.
mobility-type	(Optional) Mobility header type. Either the *mh-number* or *mh-type* argument must be used with this keyword.
mh-number	(Optional) Integer in the range from 0 to 255 representing an IPv6 mobility header type.
mh-type	(Optional) Name of a mobility header type. Possible mobility header types and their corresponding *mh-number* value are as follows: • 0—bind-refresh • 1—hoti • 2—coti • 3—hot • 4—cot • 5—bind-update • 6—bind-acknowledgment • 7—bind-error
routing	(Optional) Matches source-routed packets against the routing extension header within each IPv6 packet header.
routing-type	(Optional) Allows routing headers with a value in the type field to be matched independently. The *routing-number* argument must be used with this keyword.
routing-number	Integer in the range from 0 to 255 representing an IPv6 routing header type. Possible routing header types and their corresponding *routing-number* value are as follows: • 0—Standard IPv6 routing header • 2—Mobile IPv6 routing header

sequence *value*	(Optional) Specifies the sequence number for the access list statement. The acceptable range is from 1 to 4294967295.
time-range *name*	(Optional) Specifies the time range that applies to the deny statement. The name of the time range and its restrictions are specified by the **time-range** and **absolute** or **periodic** commands, respectively.
undetermined-transport	(Optional) Matches packets from a source for which the Layer 4 protocol cannot be determined. The **undetermined-transport** keyword is an option only if the *operator* [*port-number*] arguments are not specified.
icmp-type	(Optional) Specifies an ICMP message type for filtering ICMP packets. ICMP packets can be filtered by ICMP message type. The ICMP message type can be a number from 0 to 255, some of which include the following predefined strings and their corresponding numeric values: • 144—dhaad-request • 145—dhaad-reply • 146—mpd-solicitation • 147—mpd-advertisement
icmp-code	(Optional) Specifies an 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. The code is a number from 0 to 255.
icmp-message	(Optional) Specifies an ICMP message name for filtering ICMP packets. ICMP packets can be filtered by an ICMP message name or ICMP message type and code. The possible names are listed in the “Usage Guidelines” section.
ack	(Optional) For the TCP protocol only: acknowledgment (ACK) bit set.
established	(Optional) For the TCP protocol only: Indicates an established connection. A match occurs if the TCP datagram has the ACK or RST bits set. The nonmatching case is that of the initial TCP datagram to form a connection.

fin	(Optional) For the TCP protocol only: Fin bit set; no more data from sender.
neq {*port* | *protocol*}	(Optional) Matches only packets that are not on a given port number.
psh	(Optional) For the TCP protocol only: Push function bit set.
range {*port* | *protocol*}	(Optional) Matches only packets in the range of port numbers.
rst	(Optional) For the TCP protocol only: Reset bit set.
syn	(Optional) For the TCP protocol only: Synchronize bit set.
urg	(Optional) For the TCP protocol only: Urgent pointer bit set.

Command Default

No IPv6 access list is defined.

Command Modes

IPv6 access list configuration (config-ipv6-acl)#

Command History

Release	Modification
12.0(23)S	This command was introduced.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.4(2)T	The *icmp-type* argument was enhanced. The **dest-option-type**, **mobility**, **mobility-type**, and **routing-type** keywords were added. The *doh-number*, *doh-type*, *mh-number*, *mh-type*, and *routing-number* arguments were added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Aggregation Series Routers.

Release	Modification
12.4(20)T	The **auth** keyword was added.
12.2(33)SRE	This command was integrated into Cisco IOS Release 12.2(33)SRE.
15.2(3)T	This command was modified. Support was added for the **hbh** keyword.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **deny** (IPv6) command is similar to the **deny** (IP) 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 or to define the access list as a reflexive access list.

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

By 1default, 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**, **remark**, or **evaluate** 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.

In Cisco IOS Release 12.2(2)T or later releases, 12.0(21)ST, and 12.0(22)S, IPv6 access control lists (ACLs) are defined and their deny and permit conditions are set by using the **ipv6 access-list** command with the **deny** and **permit** keywords in global configuration mode. In Cisco IOS Release 12.0(23)S or later releases, IPv6 ACLs 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. Refer to the **ipv6 access-list** command for more information on defining IPv6 ACLs.

Note In Cisco IOS Release 12.0(23)S or later releases, 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. 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.

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.

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

The **undetermined-transport** keyword is an option only if the *operator* [*port-number*] arguments are not specified.

The following is a list of ICMP message names:

- beyond-scope
- destination-unreachable
- echo-reply
- echo-request
- header
- hop-limit
- mld-query
- mld-reduction
- mld-report
- nd-na
- nd-ns
- next-header
- no-admin
- no-route
- packet-too-big
- parameter-option
- parameter-problem
- port-unreachable
- reassembly-timeout
- renum-command
- renum-result
- renum-seq-number
- router-advertisement
- router-renumbering
- router-solicitation
- time-exceeded
- unreachable

Examples

The following example configures the IPv6 access list named toCISCO and applies the access list to outbound traffic on Ethernet interface 0. 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 Ethernet interface 0. The second deny entry in the list keeps all packets that have a source UDP port number less than 5000 from exiting out of Ethernet interface 0. The second deny also logs all matches to the console. The first permit entry in the list permits all ICMP packets to exit out of Ethernet interface 0. The second permit entry in the list permits all other traffic to exit out of Ethernet interface 0. The second permit entry is necessary because an implicit deny all condition is at the end of each IPv6 access list.

```
ipv6 access-list toCISCO
 deny tcp any any gt 5000
 deny ::/0 lt 5000 ::/0 log
 permit icmp any any
 permit any any
interface ethernet 0
 ipv6 traffic-filter toCISCO out
```

The following example shows how to allow TCP or UDP parsing although an IPsec AH is present:

```
IPv6 access list example1
    deny tcp host 2001::1 any log sequence 5
    permit tcp any any auth sequence 10
    permit udp any any auth sequence 20
```

Related Commands

Command	Description
ipv6 access-list	Defines an IPv6 access list and enters IPv6 access list configuration mode.
ipv6 traffic-filter	Filters incoming or outgoing IPv6 traffic on an interface.
permit (IPv6)	Sets permit conditions for an IPv6 access list.
show ipv6 access-list	Displays the contents of all current IPv6 access lists.

destination-glean

To enable IPv6 first-hop security binding table recovery using destination address gleaning, or to generate syslog messages about unrecognized binding table entries following a recovery, use the **destination-glean** command in IPv6 snooping configuration mode. To disable binding table recovery, use the **no** form of this command.

destination-glean {**recovery** | **log-only**} [**dhcp**]

no destination-glean

Syntax Description

recovery	Enables binding table recovery using destination address gleaning.
log-only	Generates a syslog message about unrecognized binding table entries following a recovery.
dhcp	Specifies that destination addresses should be recovered from Dynamic Host Configuration Protocol (DHCP).

Command Default

IPv6 first-hop security binding table recovery using destination address gleaning is not enabled.

Command Modes

IPv6 snooping configuration mode (config-ipv6-snooping)

Command History

Release	Modification
15.2(4)S	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

When you configure IPv6 destination guard using the **ipv6 destination-guard policy** command, you can then also configure IPv6 first-hop security binding table recovery.

The **ipv6 snooping policy** command allows you to configure a snooping policy. You can configure first-hop security binding table recovery as part of this policy. The snooping policy should then be attached to a port or VLAN using the **ipv6 snooping attach-policy** command.

If you use the **destination-glean** command with the **log-only** keyword, only a syslog message will be generated and no recovery will be attempted.

Examples

The following example shows that destination addresses should be recovered from DHCP:

```
Device(config-ipv6-snooping)# destination-glean recovery dhcp
```

The following example shows that a syslog message will be generated for all missed destination addresses following a binding table recovery:

```
Device(config-ipv6-snooping)# destination-glean log-only
```

Related Commands

Command	Description
ipv6 destination-guard policy	Configures an IPv6 destination guard policy.
ipv6 snooping policy	Enters IPv6 snooping configuration mode.

device-role

To specify the role of the device attached to the port, use the **device-role** command in neighbor discovery (ND) inspection policy configuration mode or router advertisement (RA) guard policy configuration mode.

device-role {**host**| **monitor**| **router**}

Syntax Description

host	Sets the role of the device to host.
monitor	Sets the role of the device to monitor.
router	Sets the role of the device to router.

Command Default

The device role is host.

Command Modes

ND inspection policy configuration (config-nd-inspection)

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **device-role** command specifies the role of the device attached to the port. By default, the device role is host, and therefore all the inbound router advertisement and redirect messages are blocked. If the device role is enabled using the **router** keyword, all messages (router solicitation [RS], router advertisement [RA], or redirect) are allowed on this port.

When the **router** or **monitor** keyword is used, the multicast RS messages are bridged on the port, regardless of whether limited broadcast is enabled. However, the **monitor** keyword does not allow inbound RA or redirect messages. When the **monitor** keyword is used, devices that need these messages will receive them.

Note With the introduction of Cisco IOS Release 15.2(4)S1, the trusted port has precedence over the device role for accepting RAs over a port to the router. Prior to this release, the device role router had precedence over the trusted port. The device role of the router still needs to be configured in order for the RS to be sent over the port.

Examples

The following example defines a Neighbor Discovery Protocol (NDP) policy name as policy1, places the device in ND inspection policy configuration mode, and configures the device as the host:

```
Router(config)# ipv6 nd inspection policy policy1
Router(config-nd-inspection)# device-role host
```

The following example defines an RA guard policy name as raguard1, places the device in RA guard policy configuration mode, and configures the device as the host:

```
Router(config)# ipv6 nd raguard policy raguard1
Router(config-ra-guard)# device-role host
```

Related Commands

Command	Description
ipv6 nd inspection policy	Defines the ND inspection policy name and enters ND inspection policy configuration mode.
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.

drop-unsecure

To drop messages with no or invalid options or an invalid signature, use the **drop-unsecure**command in neighbor discovery (ND) inspection policy configuration mode or or router advertisement (RA) guard policy configuration mode. To disable this function, use the **no** form of this command.

drop-unsecure

no drop-unsecure

Syntax Description

This command has no arguments or keywords.

Command Default

No ND inspection policies are configured.

Command Modes

ND inspection policy configuration (config-nd-inspection)

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **drop-unsecure** command drops messages with no or invalid Cryptographically Generated Address (CGA) options or Rivest, Shamir, and Adleman (RSA) signature as per RFC 3971, *Secure Discovery (SeND)*. However, note that messages with an RSA signature or CGA options that do not conform with or are not verified per RFC 3972, *Cryptographically Generated Addresses (CGA)*, are dropped.

Use the **drop-unsecure** command after enabling ND inspection policy configuration mode using the **ipv6 nd inspection policy** command.

Examples

The following example defines an ND policy name as policy1, places the router in ND inspection policy configuration mode, and enables the router to drop messages with invalid CGA options or an invalid RSA signature:

```
Router(config)# ipv6 nd-inspection policy policy1
Router(config-nd-inspection)# drop-unsecure
```

Related Commands

Command	Description
ipv6 nd inspection policy	Defines the ND inspection policy name and enters ND inspection policy configuration mode.
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.

enforcement

To set the enforcement level of a destination guard policy, use the **enforcement** command in destination-guard configuration mode.

enforcement {**always**| **stressed**}

Syntax Description

always	Sets the enforcement level to always.
stressed	Sets the enforcement level to forced only when the system is under stress.

Command Default

The enforcement level of a destination guard policy is set to always.

Command Modes

Destination-guard configuration (config-destguard)

Command History

Release	Modification
15.2(4)S	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Depending on the network architecture, the sources of binding table information, and the degree of change in the system, the binding table may not always have complete information about the node membership of a VLAN. The enforcement level policy element means that systems with authoritative knowledge of the VLAN membership should set the enforcement level to always. Systems with less confidence, or those with a strong desire to avoid inadvertent packet loss, should set the enforcement level to stressed.

Examples

The following example shows how to set the enforcement level to always:

```
Device(config)# ipv6 destination-guard policy destination
Device(config-destguard)# enforcement always
```

Related Commands

Command	Description
ipv6 destination-guard policy	Defines the destination guard policy.

graceful-restart

To enable the Open Shortest Path First version 3 (OSPFv3) graceful restart feature on a graceful-restart-capable router, use the **graceful-restart**command in OSPF router configuration mode. To disable graceful restart, use the **no** form of this command.

graceful-restart [**restart-interval** *interval*]

no graceful-restart

Syntax Description

restart-interval *interval*	(Optional) Graceful-restart interval in seconds. The range is from 1 to 1800, and the default is 120.

Command Default

The GR feature is not enabled on GR-capable routers.

Command Modes

OSPFv3 router configuration mode (config-router)

Command History

Release	Modification
Cisco IOS XE Release 2.1	This command was introduced.
15.0(1)M	This command was integrated into Cisco IOS Release 12.5(1)M.
12.2(33)SRE	This command was modified. It was integrated into Cisco IOS Release 12.2(33)SRE.
12.2(33)XNE	This command was modified. It was integrated into Cisco IOS Release 12.2(33)XNE.
15.1(3)S	This command was modified. The feature can be enabled in an IPv4 or IPv6 OSPFv3 process.
Cisco IOS XE Release 3.4S	This command was modified. The feature can be enabled in an IPv4 or IPv6 OSPFv3 process.
15.2(1)T	This command was modified. The feature can be enabled in an IPv4 or IPv6 OSPFv3 process.
15.1(1)SY	This command was modified. The feature can be enabled in an IPv4 or IPv6 OSPFv3 process.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **graceful-restart** command can be enabled only on GR-capable routers.

Examples

The following examples enables graceful restart mode on a GR-capable router in IPv6 and IPv4:

```
Router(config)# ospfv3 router 1
Router(config-router)# graceful-restar
```

The following examples enables graceful restart mode on a GR-capable router in IPv6 only:

```
Router(config)# ipv6 router ospf 1234
Router(config-router)# graceful-restart
```

Related Commands

Command	Description
graceful-restart helper	Enables the OSPFv3 graceful restart feature on a GR-aware router.
router ospfv3	Enables OSPFv3 router configuration mode for the IPv4 or IPv6 address family.

hop-limit

To verify the advertised hop-count limit, use the **hop-limit** command in RA guard policy configuration mode.

hop-limit {**maximum**| **minimum** } *limit*

Syntax Description

maximum *limit*	Verifies that the hop-count limit is lower than that set by the *limit* argument.
minimum *limit*	Verifies that the hop-count limit is greater than that set by the *limit* argument.

Command Default

No hop-count limit is specified.

Command Modes

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **hop-limit** command enables verification that the advertised hop-count limit is greater than or less than the value set by the *limit* argument. Configuring the **minimum** *limit* keyword and argument can prevent an attacker from setting a low hop-count limit value on the hosts to block them from generating traffic to remote destinations; that is, beyond their default router. If the advertised hop-count limit value is unspecified (which is the same as setting a value of 0), the packet is dropped.

Configuring the **maximum** *limit* keyword and argument enables verification that the advertised hop-count limit is lower than the value set by the *limit* argument. If the advertised hop-count limit value is unspecified (which is the same as setting a value of 0), the packet is dropped.

Examples

The following example shows how the command defines a router advertisement (RA) guard policy name as raguard1, places the router in RA guard policy configuration mode, and sets a minimum hop-count limit of 3:

```
Router(config)# ipv6 nd raguard policy raguard1
Router(config-ra-guard)# hop-limit minimum 3
```

Related Commands

Command	Description
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.

interval-option

To adjust the IPv6 router advertisement (RA) interval in an RA throttler policy, use the **interval-option** command in IPv6 RA throttle policy configuration mode. To reset the command to its defaults, use the **no** form of this command.

interval-option {**ignore**| **inherit**| **pass-through**| **throttle**}

Syntax Description

ignore	Interval option has no influence on throttling.
inherit	Merges the setting between target policies.
pass-through	All RAs with the interval option will be forwarded.
throttle	All RAs with the interval option will be throttled.

Command Default

Pass-through

Command Modes

IPv6 RA throttle policy configuration mode (config-nd-ra-throttle)

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

The **interval-option** command configures an interval option for an RA throttler policy. An interval option, as defined by RFC 6275, is used in RA messages to advertise the interval at which the sending device sends unsolicited multicast RAs.

Examples

```
Device(config)# ipv6 nd ra-throttle policy policy1
Device(config-nd-ra-throttle)# interval-option inherit
```

ipv6 access-list

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

ipv6 access-list *access-list-name*

no ipv6 access-list *access-list-name*

Syntax Description

access-list-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

Global configuration

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.0(23)S	Support for IPv6 address configuration mode and extended access list functionality (the filtering of traffic based on IPv6 option headers and optional, upper-layer protocol type information) was added. Additionally, the following keywords and arguments were moved from global configuration mode to IPv6 access list configuration mode: **permit**, **deny,** *source-ipv6-prefix* / *prefix-length*, **any**, *destination-ipv6-prefix* / *prefix-length*, **priority**. See the "Usage Guidelines" section for more details.
12.2(13)T	Support for IPv6 address configuration mode and extended access list functionality (the filtering of traffic based on IPv6 option headers and optional, upper-layer protocol type information) was added. Additionally, the following keywords and arguments were moved from global configuration mode to IPv6 access list configuration mode: **permit**, **deny,** *source-ipv6-prefix* / *prefix-length*, **any**, *destination-ipv6-prefix* / *prefix-length*, **priority**. See the "Usage Guidelines" section for more details.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.

Release	Modification
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	Duplicate remark statements can no longer be configured from the IPv6 access control list.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 series devices.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

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

In Cisco IOS Release 12.2(2)T or later releases, 12.0(21)ST, and 12.0(22)S, standard IPv6 access control list (ACL) functionality is used for basic traffic filtering functions--traffic filtering is based on source and destination addresses, inbound and outbound to a specific interface, and with an implicit deny statement at the end of each access list (functionality similar to standard ACLs in IPv4). IPv6 ACLs are defined and their deny and permit conditions are set by using the **ipv6 access-list**command with the **deny** and **permit** keywords in global configuration mode.

In Cisco IOS Release 12.0(23)S or later releases, the standard IPv6 ACL functionality is extended to support--in addition to traffic filtering based on source and destination addresses--filtering of traffic based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control (functionality similar to extended ACLs in IPv4). IPv6 ACLs 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 device in IPv6 access list configuration mode--the device prompt changes to Device(config-ipv6-acl)#. From IPv6 access list configuration mode, permit and deny conditions can be set for the defined IPv6 ACL.

Note IPv6 ACLs are defined by a unique name (IPv6 does not support numbered ACLs). An IPv4 ACL and an IPv6 ACL cannot share the same name.

In Cisco IOS Release 12.0(23)S or later releases, and 12.2(11)S or later releases, for backward compatibility, the **ipv6 access-list**command with the **deny** and **permit** keywords in global configuration mode is still supported; however, an IPv6 ACL defined with deny and permit conditions in global configuration mode is translated to IPv6 access list configuration mode.

Refer to the deny (IPv6) and permit (IPv6) commands for more information on filtering IPv6 traffic based on IPv6 option headers and optional, upper-layer protocol type information. See the "Examples" section for an example of a translated IPv6 ACL configuration.

Note In Cisco IOS Release 12.0(23)S or later releases, 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. 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.

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

Use the **ipv6 traffic-filter** interface configuration command with the *access-list-name* argument to apply an IPv6 ACL to an IPv6 interface. Use the **ipv6 access-class** line configuration command with the *access-list-name* argument to apply an IPv6 ACL to incoming and outgoing IPv6 virtual terminal connections to and from the device.

Note An IPv6 ACL applied to an interface with the **ipv6 traffic-filter** command filters traffic that is forwarded, not originated, by the device.

Note When using this command to modify an ACL that is already associated with a bootstrap router (BSR) candidate rendezvous point (RP) (see the **ipv6 pim bsr candidate rp** command) or a static RP (see the **ipv6 pim rp-address** command), any added address ranges that overlap the PIM SSM group address range (FF3x::/96) are ignored. A warning message is generated and the overlapping address ranges are added to the ACL, but they have no effect on the operation of the configured BSR candidate RP or static RP commands.

In Cisco IOS Release 12.2(33)SXH and subsequent Cisco IOS SX releases, duplicate remark statements can no longer be configured from the IPv6 access control list. Because each remark statement is a separate entity, each one is required to be unique.

Examples

The following example is from a device running Cisco IOS Release 12.0(23)S or later releases. The example configures the IPv6 ACL list named list1 and places the device in IPv6 access list configuration mode.

```
Device(config)# ipv6 access-list list1
Device(config-ipv6-acl)#
```

The following example is from a device running Cisco IOS Release 12.2(2)T or later releases, 12.0(21)ST, or 12.0(22)S. The example configures the IPv6 ACL named list2 and applies the ACL to outbound traffic on Ethernet interface 0. 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 Ethernet interface 0. The second entry in the ACL permits all other traffic to exit out of Ethernet interface 0. The second entry is necessary because an implicit deny all condition is at the end of each IPv6 ACL.

```
Device(config)# ipv6 access-list list2 deny FEC0:0:0:2::/64 any
Device(config)# ipv6 access-list list2 permit any any
Device(config)# interface ethernet 0
Device(config-if)# ipv6 traffic-filter list2 out
```

If the same configuration was entered on a device running Cisco IOS Release 12.0(23)S or later releases, the configuration would be translated into IPv6 access list configuration mode as follows:

```
ipv6 access-list list2
  deny FEC0:0:0:2::/64 any
  permit ipv6 any any
interface ethernet 0
 ipv6 traffic-filter 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 IPv6 ACLs defined on a device running Cisco IOS Release 12.2(2)T or later releases, 12.0(21)ST, or 12.0(22)S that rely on the implicit deny condition or specify a **deny any any** statement to filter traffic should contain **permit** statements for link-local and multicast addresses to avoid the filtering of protocol packets (for example, packets associated with the neighbor discovery protocol). Additionally, IPv6 ACLs that use **deny** statements to filter traffic should use a **permit any any** statement as the last statement in the list.

Note An IPv6 device will 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).

Related Commands

Command	Description
deny (IPv6)	Sets deny conditions for an IPv6 access list.
ipv6 access-class	Filters incoming and outgoing connections to and from the device based on an IPv6 access list.
ipv6 pim bsr candidate rp	Configures the candidate RP to send PIM RP advertisements to the BSR.
ipv6 pim rp-address	Configure the address of a PIM RP for a particular group range.
ipv6 traffic-filter	Filters incoming or outgoing IPv6 traffic on an interface.
permit (IPv6)	Sets permit conditions for an IPv6 access list.
show ipv6 access-list	Displays the contents of all current IPv6 access lists.

ipv6 address

To configure an IPv6 address based on an IPv6 general prefix and enable IPv6 processing on an interface, 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*| *prefix-name sub-bits/prefix-length*}

no ipv6 address {*ipv6-address/prefix-length*| *prefix-name sub-bits/prefix-length*}

Syntax Description

ipv6-address	The 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.
prefix-name	A general prefix, which specifies the leading bits of the network to be configured on the interface.
sub-bits	The subprefix bits and host bits of the address to be concatenated with the prefixes provided by the general prefix specified with the *prefix-name* argument. The *sub-bits*argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.

Command Default

No IPv6 addresses are defined for any interface.

Command Modes

Interface configuration

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Release	Modification
Cisco IOS XE Release 2.1	This command was integrated into Cisco ASR 1000 Series devices.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.2(2)SNG	This command was implemented on the Cisco ASR 901 Series Aggregation Services devices.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 address** command allows multiple IPv6 addresses to be configured on an interface in various different ways, with varying options. The most common way is to specify the IPv6 address with the prefix length.

Addresses may also be defined using the general prefix mechanism, which separates the aggregated IPv6 prefix bits from the subprefix and host bits. In this case, the leading bits of the address are defined in a general prefix, which is globally configured or learned (for example, through use of Dynamic Host Configuration Protocol-Prefix Delegation (DHCP-PD)), and then applied using the *prefix-name* argument. The subprefix bits and host bits are defined using the *sub-bits* argument.

Using the **no ipv6 address autoconfig** command without arguments removes all IPv6 addresses from an interface.

IPv6 link-local addresses must be configured and IPv6 processing must be enabled on an interface by using the **ipv6 address link-local** command.

Examples

The following example shows how to enable IPv6 processing on the interface and configure an address based on the general prefix called my-prefix and the directly specified bits:

```
Device(config-if) ipv6 address my-prefix 0:0:0:7272::72/64
```

Assuming the general prefix named my-prefix has the value of 2001:DB8:2222::/48, then the interface would be configured with the global address 2001:DB8:2222:7272::72/64.

Related Commands

Command	Description
ipv6 address anycast	Configures an IPv6 anycast address and enables IPv6 processing on an interface.
ipv6 address eui-64	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	Configures an IPv6 link-local address for an interface and enables IPv6 processing on the interface.
ipv6 unnumbered	Enables IPv6 processing on an interface without assigning an explicit IPv6 address to the interface.
no ipv6 address autoconfig	Removes all IPv6 addresses from an interface.

Command	Description
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.

ipv6 address anycast

To configure an IPv6 anycast address and enable IPv6 processing on an interface, use the **ipv6 address anycast**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* **anycast**

no ipv6 address [*ip6-prefix/prefix-length* **anycast**]

Syntax Description

ipv6-prefix	The IPv6 network assigned to the interface. 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 comprise the prefix (the network portion of the address). A slash mark must precede the decimal value.

Command Default

No IPv6 addresses are defined for any interface.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.2(25)S	This command was integrated into Cisco IOS Release 12.2(25)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Using the **no ipv6 address** command without arguments removes all manually configured IPv6 addresses from an interface.

Examples

The following example shows how to enable IPv6 processing on the interface, assign the prefix 2001:0DB8:1:1::/64 to the interface, and configure the IPv6 anycast address 2001:0DB8:1:1:FFFF:FFFF:FFFF:FFFE:

```
ipv6 address 2001:0DB8:1:1:FFFF:FFFF:FFFF:FFFE/64 anycast
```

Related Commands

Command	Description
ipv6 address eui-64	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	Configures an IPv6 link-local address for an interface and enables IPv6 processing on the interface.
ipv6 unnumbered	Enables IPv6 processing on an interface without assigning an explicit IPv6 address to the interface.
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.

ipv6 address autoconfig

To enable automatic configuration of IPv6 addresses using stateless autoconfiguration on an interface and enable IPv6 processing on the interface, use the **ipv6 address autoconfig** command in interface configuration mode. To remove the address from the interface, use the **no** form of this command.

ipv6 address autoconfig [default]

no ipv6 address autoconfig

Syntax Description

default	(Optional) If a default device is selected on this interface, the **default** keyword causes a default route to be installed using that default device. The **default** keyword can be specified only on one interface.

Command Default

No IPv6 address is defined for the interface.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(13)T	This command was introduced.
12.2(33)SRE	This command was integrated into Cisco IOS Release 12.2(33)SRE.
Cisco IOS XE Release 2.5	This command was integrated into Cisco IOS XE Release 2.5.
12.2(33)XNE	This command was integrated into Cisco IOS Release 12.2(33)XNE.
15.1(2)SNG	This command was implemented on the Cisco ASR 901 Series Aggregation Services devices.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 address autoconfig** command causes the device to perform IPv6 stateless address auto-configuration to discover prefixes on the link and then to add the EUI-64 based addresses to the interface. Addresses are configured depending on the prefixes received in Router Advertisement (RA) messages.

Using the **no ipv6 address autoconfig** command without arguments removes all IPv6 addresses from an interface.

Examples

The following example assigns the IPv6 address automatically:

```
Device(config)# interface ethernet 0
Device(config-if)# ipv6 address autoconfig
```

Related Commands

Command	Description
ipv6 address eui-64	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	Configures an IPv6 link-local address for an interface and enables IPv6 processing on the interface.
ipv6 unnumbered	Enables IPv6 processing on an interface without assigning an explicit IPv6 address to the interface.
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.

ipv6 address dhcp

To acquire an IPv6 address on an interface from the Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server, use the **ipv6 address dhcp** command in the interface configuration mode. To remove the address from the interface, use the **no** form of this command.

ipv6 address dhcp [rapid-commit]

no ipv6 address dhcp

Syntax Description

rapid-commit	(Optional) Allows the two-message exchange method for address assignment.

Command Default

No IPv6 addresses are acquired from the DHCPv6 server.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.4(24)T	This command was introduced.
12.2(33)SRE	This command was integrated into Cisco IOS Release 12.2(33)SRE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 address dhcp** interface configuration command allows any interface to dynamically learn its IPv6 address by using DHCP.

The **rapid-commit** keyword enables the use of the two-message exchange for address allocation and other configuration. If it is enabled, the client includes the rapid-commit option in a solicit message.

Examples

The following example shows how to acquire an IPv6 address and enable the rapid-commit option:

```
Router(config)# interface fastethernet 0/0
Router(config-if)# ipv6 address dhcp
rapid-commit
```

You can verify your settings by using the **show ipv6 dhcp interface** command in privileged EXEC mode.

Related Commands

Command	Description
show ipv6 dhcp interface	Displays DHCPv6 interface information.

ipv6 address eui-64

To configure an IPv6 address for an interface and enables IPv6 processing on the interface using an EUI-64 interface ID in the low order 64 bits of the address, use the **ipv6 address eui-64**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**

no ipv6 address [*ip v6-prefix/prefix-length* **eui-64**]

Syntax Description

ipv6-prefix	The IPv6 network assigned to the interface. 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 comprise the prefix (the network portion of the address). A slash mark must precede the decimal value.

Command Default

No IPv6 address is defined for the interface.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Series Routers.

Release	Modification
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

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 software detects another host using one of its IPv6 addresses, it will display an error message on the console.

Examples

The following example assigns IPv6 address 2001:0DB8:0:1::/64 to Ethernet interface 0 and specifies an EUI-64 interface ID in the low order 64 bits of the address:

```
Router(config)# interface ethernet 0
Router(config-if)# ipv6 address 2001:0DB8:0:1::/64 eui-64
```

Related Commands

Command	Description
ipv6 address link-local	Configures an IPv6 link-local address for an interface and enables IPv6 processing on the interface.
ipv6 unnumbered	Enables IPv6 processing on an interface without assigning an explicit IPv6 address to the interface.
show ipv6 interface	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/prefix-length* **link-local [cga]**

no ipv6 address [*ipv6-address/prefix-length* **link-local**]

Syntax Description

ipv6-address	The IPv6 address assigned to the interface. 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 comprise the prefix (the network portion of the address). A slash mark must precede the decimal value.
link-local	Specifies a link-local address. The *ipv6-address* specified with this command overrides the link-local address that is automatically generated for the interface.
cga	(Optional) Specifies the CGA interface identifier.

Command Default

No IPv6 address is defined for the interface.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.

Release	Modification
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Series Routers.
12.4(24)T	The **cga** keyword was added
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Using the **no ipv6 address command** without arguments removes all manually configured IPv6 addresses from an interface.

If the Cisco software detects another host using one of its IPv6 addresses, it will display 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.

Examples

The following example assigns FE80::260:3EFF:FE11:6770 as the link-local address for Ethernet interface 0:

```
interface ethernet 0
 ipv6 address FE80::260:3EFF:FE11:6770 link-local
```

Related Commands

Command	Description
ipv6 address eui-64	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 unnumbered	Enables IPv6 processing on an interface without assigning an explicit IPv6 address to the interface.
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.

ipv6 cef

To enable Cisco Express Forwarding for IPv6, use the **ipv6 cef** command in global configuration mode. To disable Cisco Express Forwarding for IPv6, use the **no** form of this command.

ipv6 cef

no ipv6 cef

Syntax Description

This command has no arguments or keywords.

Command Default

Cisco Express Forwarding for IPv6 is disabled by default.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.0(22)S	This command was introduced.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was implemented on Cisco ASR 1000 Series Aggregation Services Routers.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 cef** command is similar to the **ip cef** command, except that it is IPv6-specific.

The **ipv6 cef** command is not available on the Cisco 12000 series Internet routers because this distributed platform operates only in distributed Cisco Express Forwarding for IPv6 mode.

Note The **ipv6 cef**command is not supported in interface configuration mode.

Note Some distributed architecture platforms, such as the Cisco 7500 series routers, support both Cisco Express Forwarding for IPv6 and distributed Cisco Express Forwarding for IPv6. When Cisco Express Forwarding for IPv6 is configured on distributed platforms, Cisco Express Forwarding switching is performed by the Route Processor (RP).

Note You must enable Cisco Express Forwarding for IPv4 by using the **ip cef** global configuration command before enabling Cisco Express Forwarding for IPv6 by using the **ipv6 cef** global configuration command.

Cisco Express Forwarding for IPv6 is advanced Layer 3 IP switching technology that functions the same and offer the same benefits as Cisco Express Forwarding for IPv4. Cisco Express Forwarding for IPv6 optimizes network performance and scalability for networks with dynamic, topologically dispersed traffic patterns, such as those associated with web-based applications and interactive sessions.

Examples

The following example enables standard Cisco Express Forwarding for IPv4 operation and then standard Cisco Express Forwarding for IPv6 operation globally on the router.

```
ip cef
ipv6 cef
```

Related Commands

Command	Description
ip route-cache	Controls the use of high-speed switching caches for IP routing.
ipv6 cef accounting	Enables Cisco Express Forwarding for IPv6 and distributed Cisco Express Forwarding for IPv6 network accounting.
ipv6 cef distributed	Enables distributed Cisco Express Forwarding for IPv6.
show cef	Displays which packets the line cards dropped or displays which packets were not express-forwarded.
show ipv6 cef	Displays entries in the IPv6 FIB.

ipv6 cef accounting

To enable Cisco Express Forwarding for IPv6 and distributed Cisco Express Forwarding for IPv6 network accounting, use the **ipv6 cef accounting**command in global configuration mode or interface configuration mode. To disable Cisco Express Forwarding for IPv6 network accounting, use the **no**form of this command.

ipv6 cef accounting *accounting-types*

no ipv6 cef accounting *accounting-types*

Specific Cisco Express Forwarding Accounting Information Through Interface Configuration Mode

ipv6 cef accounting non-recursive {**external**| **internal**}

no ipv6 cef accounting non-recursive {**external**| **internal**}

Syntax Description

accounting-types	The *accounting-types* argument must be replaced with at least one of the following keywords. Optionally, you can follow this keyword by any or all of the other keywords, but you can use each keyword only once. • **load-balance-hash** --Enables load balancing hash bucket counters. • **non-recursive** --Enables accounting through nonrecursive prefixes. • **per-prefix** --Enables express forwarding of the collection of the number of packets and bytes to a destination (or prefix). • **prefix-length** --Enables accounting through prefix length.
non-recursive	Enables accounting through nonrecursive prefixes. This keyword is optional when used in global configuration mode after another keyword is entered. See the *accounting-types* argument.
external	Counts input traffic in the nonrecursive external bin.
internal	Counts input traffic in the nonrecursive internal bin.

Command Default

Cisco Express Forwarding for IPv6 network accounting is disabled by default.

Command Modes

Global configuration (config) Interface configuration (config-if)

Command History

Release	Modification
12.0(22)S	This command was introduced.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(25)S	The **non-recursive**and **load-balance-hash**keywords were added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Series Routers.
12.4(20)T	This command was integrated into Cisco IOS Release 12.4(20)T.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 cef accounting**command is similar to the **ip cef accounting**command, except that it is IPv6-specific.

Configuring Cisco Express Forwarding for IPv6 network accounting enables you to collect statistics on Cisco Express Forwarding for IPv6 traffic patterns in your network.

When you enable network accounting for Cisco Express Forwarding for IPv6 by using the **ipv6 cef accounting**command in global configuration mode, accounting information is collected at the Route Processor (RP) when Cisco Express Forwarding for IPv6 mode is enabled and at the line cards when distributed Cisco Express Forwarding for IPv6 mode is enabled. You can then display the collected accounting information using the **show ipv6 cef** EXEC command.

For prefixes with directly connected next hops, the **non-recursive** keyword enables express forwarding of the collection of packets and bytes through a prefix. This keyword is optional when this command is used in global configuration mode after you enter another keyword on the **ipv6 cef accounting**command.

This command in interface configuration mode must be used in conjunction with the global configuration command. The interface configuration command allows a user to specify two different bins (internal or external) for the accumulation of statistics. The internal bin is used by default. The statistics are displayed through the **show ipv6 cef detail**command.

Per-destination load balancing uses a series of 16 hash buckets into which the set of available paths are distributed. A hash function operating on certain properties of the packet is applied to select a bucket that contains a path to use. The source and destination IP addresses are the properties used to select the bucket for per-destination load balancing. Use the **load-balance-hash** keyword with the **ipv6 cef accounting** command to enable per-hash-bucket counters. Enter the **show ipv6 cef** *prefix* **internal** command to display the per-hash-bucket counters.

Examples

The following example enables the collection of Cisco Express Forwarding for IPv6 accounting information for prefixes with directly connected next hops:

```
Router(config)# ipv6 cef accounting non-recursive
```

Related Commands

Command	Description
ip cef accounting	Enable Cisco Express Forwarding network accounting (for IPv4).
show cef	Displays information about packets **forwarded by Cisco Express Forwarding.**
show ipv6 cef	Displays entries in the IPv6 FIB.

ipv6 cef distributed

To enable distributed Cisco Express Forwarding for IPv6, use the **ipv6 cef distributed**command in global configuration mode. To disable Cisco Express Forwarding for IPv6, use the **no** form of this command.

ipv6 cef distributed

no ipv6 cef distributed

Syntax Description

This command has no arguments or keywords.

Command Default

Distributed Cisco Express Forwarding for IPv6 is disabled on the Cisco 7500 series routers and enabled on the Cisco 12000 series Internet routers.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.0(22)S	This command was introduced.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was implemented on Cisco ASR 1000 Series Aggregation Services Routers.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 cef distributed**command is similar to the **ip cef distributed**command, except that it is IPv6-specific.

Enabling distributed Cisco Express Forwarding for IPv6 globally on the router by using the **ipv6 cef distributed**in global configuration mode distributes the Cisco Express Forwarding processing of IPv6 packets from the Route Processor (RP) to the line cards of distributed architecture platforms.

Note The **ipv6 cef distributed** command is not supported on the Cisco 12000 series Internet routers because distributed Cisco Express Forwarding for IPv6 is enabled by default on this platform.

Note To forward distributed Cisco Express Forwarding for IPv6 traffic on the router, configure the forwarding of IPv6 unicast datagrams globally on your router by using the **ipv6 unicast-routing** global configuration command, and configure an IPv6 address and IPv6 processing on an interface by using the **ipv6 address** interface configuration command.

Note You must enable distributed Cisco Express Forwarding for IPv4 by using the **ip cef distributed**global configuration command before enabling distributed Cisco Express Forwarding for IPv6 by using the **ipv6 cef distributed**global configuration command.

Cisco Express Forwarding is advanced Layer 3 IP switching technology. Cisco Express Forwarding optimizes network performance and scalability for networks with dynamic, topologically dispersed traffic patterns, such as those associated with web-based applications and interactive sessions.

Examples

The following example enables distributed Cisco Express Forwarding for IPv6 operation:

```
ipv6 cef distributed
```

Related Commands

Command	Description
ip route-cache	Controls the use of high-speed switching caches for IP routing.
show ipv6 cef	Displays entries in the IPv6 FIB.

ipv6-i1

ipv6 dhcp guard attach-policy

To attach a Dynamic Host Configuration Protocol for IPv6 (DHCPv6) guard policy, use the **ipv6 dhcp guard attach-policy** command in interface configuration or VLAN configuration mode. To unattach the DHCPv6 guard policy, use the **no** form of this command.

Syntax Available In Interface Configuration Mode

ipv6 dhcp guard [**attach-policy** [*policy-name*]] [**vlan** {**add**| **all**| **except**| **none**| **remove**} *vlan-id* [... *vlan-id*]]

no ipv6 dhcp guard [**attach-policy** [*policy-name*]] [**vlan** {**add**| **all**| **except**| **none**| **remove**} *vlan-id* [... *vlan-id*]]

Syntax Available In VLAN Configuration Mode

ipv6 dhcp guard attach-policy [*policy-name*]

no ipv6 dhcp guard attach-policy [*policy-name*]

Syntax Description

policy-name	(Optional) DHCPv6 guard policy name.
vlan	(Optional) Specifies that the DHCPv6 policy is to be attached to a VLAN.
add	(Optional) Attaches a DHCPv6 guard policy to the specified VLAN(s).
all	(Optional) Attaches a DHCPv6 guard policy to all VLANs.
except	(Optional) Attaches a DHCPv6 guard policy to all VLANs except the specified VLAN(s).
none	(Optional) Attaches a DHCPv6 guard policy to none of the specified VLAN(s).
remove	(Optional) Removes a DHCPv6 guard policy from the specified VLAN(s).
vlan-id	(Optional) Identity of the VLAN(s) to which the DHCP guard policy applies.

Command Default

No DHCPv6 guard policy is attached.

Command Modes

Interface configuration (config-if)

VLAN configuration (config-vlan)

Command History

Release	Modification
15.2(4)S	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

This command allows you to attach a DHCPv6 policy to an interface or to one or more VLANs. DHCPv6 guard policies can be used to block reply and advertisement messages that come from unauthorized DHCP servers and relay agents that forward DHCP packets from servers to clients. Client messages or messages sent by relay agents from clients to servers are not blocked.

Examples

The following example shows how to attach a DHCPv6 guard policy to an interface:

```
Router> enable
Router# configure terminal
Router(config)# interface GigabitEthernet 0/2/0
Router# switchport
Router(config-if)# ipv6 dhcp guard attach-policy pol1 vlan add 1
```

Related Commands

Command	Description
ipv6 dhcp guard policy	Defines the DHCPv6 guard policy name.
show ipv6 dhcp guard policy	Displays DHCPv6 guard policy information.

ipv6 dhcp guard policy

To define a Dynamic Host Configuration Protocol for IPv6 (DHCPv6) guard policy name, use the **ipv6 dhcp guard policy** command in global configuration mode. To remove the DHCPv6 guard policy name, use the **no** form of this command.

ipv6 dhcp guard policy [*policy-name*]

no ipv6 dhcp guard policy [*policy-name*]

Syntax Description

policy-name	(Optional) DHCPv6 guard policy name.

Command Default

No DHCPv6 guard policy name is defined.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.2(4)S	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

This command allows you to enter DHCPv6 guard configuration mode. DHCPv6 guard policies can be used to block reply and advertisement messages that come from unauthorized DHCP servers and relay agents that forward DHCP packets from servers to clients. Client messages or messages sent by relay agents from clients to servers are not blocked.

Examples

The following example shows how to define a DHCPv6 guard policy name:

```
Router> enable
Router# configure terminal
Router(config)# ipv6 dhcp guard policy policy1
```

Related Commands

Command	Description
show ipv6 dhcp guard policy	Displays DHCPv6 guard policy information.

ipv6 dhcp ping packets

To specify the number of packets a Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server sends to a pool address as part of a ping operation, use the **ipv6 dhcp ping packets** command in global configuration mode. To prevent the server from pinging pool addresses, use the **no** form of this command.

ipv6 dhcp ping packets *number*

ipv6 dhcp ping packets

Syntax Description

number	The number of ping packets sent before the address is assigned to a requesting client. The valid range is from 0 to 10.

Command Default

No ping packets are sent before the address is assigned to a requesting client.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.4(24)T	This command was introduced.
12.2(33)SRE	This command was integrated into Cisco IOS Release 12.2(33)SRE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The DHCPv6 server pings a pool address before assigning the address to a requesting client. If the ping is unanswered, the server assumes, with a high probability, that the address is not in use and assigns the address to the requesting client.

Setting the *number* argument to 0 turns off the DHCPv6 server ping operation

Examples

The following example specifies four ping attempts by the DHCPv6 server before further ping attempts stop:

```
Router(config)# ipv6 dhcp ping packets 4
```

Related Commands

Command	Description
clear ipv6 dhcp conflict	Clears an address conflict from the DHCPv6 server database.

Command	Description
show ipv6 dhcp conflict	Displays address conflicts found by a DHCPv6 server, or reported through a DECLINE message from a client.

ipv6 dhcp server

To enable Dynamic Host Configuration Protocol (DHCP) for IPv6 service on an interface, use the **ipv6 dhcp server** in interface configuration mode. To disable DHCP for IPv6 service on an interface, use the **no** form of this command.

ipv6 dhcp server [*poolname*| **automatic**] **[rapid-commit]** [**preference** *value*] **[allow-hint]**

no ipv6 dhcp server

Syntax Description

poolname	(Optional) User-defined name for the local prefix pool. The pool name can be a symbolic string (such as "Engineering") or an integer (such as 0).
automatic	(Optional) Enables the server to automatically determine which pool to use when allocating addresses for a client.
rapid-commit	(Optional) Allows the two-message exchange method for prefix delegation.
preference *value*	(Optional) Specifies the preference value carried in the preference option in the advertise message sent by the server. The range is from 0 to 255. The preference value defaults to 0.
allow-hint	(Optional) Specifies whether the server should consider delegating client suggested prefixes. By default, the server ignores client-hinted prefixes.

Command Default

DHCP for IPv6 service on an interface is disabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.3(4)T	This command was introduced.
12.2(18)SXE	This command was integrated into Cisco IOS Release 12.2(18)SXE.
12.4(24)T	The **automatic** keyword was added.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.

Release	Modification
12.2(33)SRE	This command was integrated into Cisco IOS Release 12.2(33)SRE.
12.2(33)XNE	This command was integrated into Cisco IOS Release 12.2(33)XNE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 dhcp server** command enables DHCP for IPv6 service on a specified interface using the pool for prefix delegation and other configuration through that interface.

The **automatic** keyword enables the system to automatically determine which pool to use when allocating addresses for a client. When an IPv6 DHCP packet is received by the server, the server determines if it was received from a DHCP relay or if it was directly received from the client. If the packet was received from a relay, the server verifies the link-address field inside the packet associated with the first relay that is closest to the client. The server matches this link address against all address prefix and link-address configurations in IPv6 DHCP pools to find the longest prefix match. The server selects the pool associated with the longest match.

If the packet was directly received from the client, the server performs this same matching, but it uses all the IPv6 addresses configured on the incoming interface when performing the match. Once again, the server selects the longest prefix match.

The **rapid-commit** keyword enables the use of the two-message exchange for prefix delegation and other configuration. If a client has included a rapid commit option in the solicit message and the **rapid-commit** keyword is enabled for the server, the server responds to the solicit message with a reply message.

If the **preference** keyword is configured with a value other than 0, the server adds a preference option to carry the preference value for the advertise messages. This action affects the selection of a server by the client. Any advertise message that does not include a preference option is considered to have a preference value of 0. If the client receives an advertise message that includes a preference option with a preference value of 255, the client immediately sends a request message to the server from which the advertise message was received.

If the **allow-hint** keyword is specified, the server will delegate a valid client-suggested prefix in the solicit and request messages. The prefix is valid if it is in the associated local prefix pool and it is not assigned to a device. If the **allow-hint** keyword is not specified, a hint is ignored and a prefix is delegated from the free list in the pool.

The DHCP for IPv6 client, server, and relay functions are mutually exclusive on an interface. When one of these functions is already enabled and a user tries to configure a different function on the same interface, one of the following messages is displayed:

```
Interface is in DHCP client mode
Interface is in DHCP server mode
Interface is in DHCP relay mode
```

Examples

The following example enables DHCP for IPv6 for the local prefix pool named server1:

```
Router(config-if)# ipv6 dhcp server server1
```

Related Commands

Command	Description
ipv6 dhcp pool	Configures a DHCP for IPv6 pool and enters DHCP for IPv6 pool configuration mode.
show ipv6 dhcp interface	Displays DHCP for IPv6 interface information.

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 interface 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

no ipv6 enable

Syntax Description

This command has no arguments or keywords.

Command Default

IPv6 is disabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.
15.2(2)SNG	This command was implemented on the Cisco ASR 901 Series Aggregation Services devices.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

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.

Examples

The following example enables IPv6 processing on Ethernet interface 0/0:

```
Device(config)# interface ethernet 0/0
Device(config-if)# ipv6 enable
```

Related Commands

Command	Description
ipv6 address link-local	Configures an IPv6 link-local address for an interface and enables IPv6 processing on the interface.
ipv6 address eui-64	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 unnumbered	Enables IPv6 processing on an interface without assigning an explicit IPv6 address to the interface.
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.

ipv6 host

To define a static host name-to-address mapping in the host name cache, use the **ipv6 host**command in global configuration mode. To remove the host name-to-address mapping, use the no form of this command.

ipv6 host *name* [*port*] *ipv6-address*

no ipv6 host *name*

Syntax Description

name	Name of the IPv6 host. The first character can be either a letter or a number. If you use a number, the operations you can perform are limited.
port	(Optional) The default Telnet port number for the associated IPv6 addresses.
ipv6-address	Associated IPv6 address. You can bind up to four addresses to a host name.

Command Default

Static host name-to-address mapping in the host name cache is not defined. The default Telnet port is 23.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The first character of the *name* variable can be either a letter or a number. If you use a number, the operations you can perform (such as **ping**) are limited.

Examples

The following example defines two static mappings:

```
Device(config)# ipv6 host cisco-sj 2001:0DB8:1::12
Device(config)# ipv6 host cisco-hq 2002:C01F:768::1 2001:0DB8:1::12
```

Related Commands

Command	Description
show hosts	Displays the default domain name, the style of name lookup service, a list of name server hosts, and the cached list of host names and addresses.

ipv6 icmp error-interval

To configure the interval and bucket size for IPv6 Internet Control Message Protocol (ICMP) error messages, use the **ipv6 icmp error-interval** command in global configuration 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	The time interval between tokens being placed in the bucket. The acceptable range is from 0 to 2147483647 with a default of 100 milliseconds.
bucketsize	(Optional) The maximum number of tokens stored in the bucket. The acceptable range is from 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. The time interval between tokens placed in the bucket is 100 milliseconds. The maximum number of tokens stored in the bucket is 10.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.2(8)T	Support for IPv6 ICMP rate limiting was extended to use token buckets.
12.0(21)ST	This command, without the extension to use token buckets, was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command, without the extension to use token buckets, was integrated into Cisco IOS Release 12.0(22)S.
12.0(23)S	This command, with the support for IPv6 ICMP rate limiting extended to use token buckets, was integrated into Cisco IOS Release 12.0(23)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.

Release	Modification
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.
15.2(2)SNG	This command was implemented on the Cisco ASR 901 Series Aggregation Services devices.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Use the **ipv6 icmp error-interval** command to limit the rate at which IPv6 ICMP error messages are sent. 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 arriving in the bucket. The optional *bucketsize* argument is used to define the maximum number of tokens allowed in the bucket. Tokens are removed from the bucket when IPv6 ICMP error messages are sent, which means that if the *bucketsize* 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** command to display IPv6 ICMP rate-limited counters.

Examples

The following example shows an interval of 50 milliseconds and a bucket size of 20 tokens being configured for IPv6 ICMP error messages:

```
ipv6 icmp error-interval 50 20
```

Related Commands

Command	Description
show ipv6 traffic	Displays statistics about IPv6 traffic.

ipv6 nd cache expire

To configure the length of time before an IPv6 neighbor discovery (ND) cache entry expires, use the **ipv6 nd cache expire** command in interface configuration mode. To remove this configuration, use the **no** form of this command.

ipv6 nd cache expire *expire-time-in-seconds* **[refresh]**

no ipv6 nd cache expire *expire-time-in-seconds* **[refresh]**

Syntax Description

expire-time-in-seconds	The time range is from 1 through 65536 seconds. The default is 14400 seconds, or 4 hours.
refresh	(Optional) Automatically refreshes the ND cache entry.

Command Default

This expiration time is 14400 seconds (4 hours)

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(33)SXI7	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

By default, an ND cache entry is expired and deleted if it remains in the STALE state for 14,400 seconds, or 4 hours. The **ipv6 nd cache expire** command allows the user to vary the expiry time and to trigger autorefresh of an expired entry before the entry is deleted.

When the **refresh** keyword is used, an ND cache entry is autorefreshed. The entry moves into the DELAY state and the neighbor unreachability detection (NUD) process occurs, in which the entry transitions from the DELAY state to the PROBE state after 5 seconds. When the entry reaches the PROBE state, a neighbor solicitation (NS) is sent and then retransmitted as per the configuration.

Examples

The following example shows that the ND cache entry is configured to expire in 7200 seconds, or 2 hours:

```
Router(config-if)# ipv6 nd cache expire 7200
```

ipv6 nd inspection

To apply the Neighbor Discovery Protocol (NDP) Inspection feature, use the **ipv6 nd inspection** command in interface configuration mode. To remove the NDP Inspection feature, use the **no** form of this command.

ipv6 nd inspection [**attach-policy** [*policy-name*] | **vlan** {**add** | **except** | **none** | **remove** | **all**} **vlan** *vlan-id*]]

no ipv6 nd inspection

Syntax Description

attach-policy	(Optional) Attaches an NDP Inspection policy.
policy-name	(Optional) The NDP Inspection policy name.
vlan	(Optional) Applies the ND Inspection feature to a VLAN on the interface.
add	(Optional) Adds a VLAN to be inspected.
except	(Optional) Inspects all VLANs except the one specified.
none	(Optional) Specifies that no VLANs are inspected.
remove	(Optional) Removes the specified VLAN from NDP inspection.
all	(Optional) Inspects NDP traffic from all VLANs on the port.
vlan-id	(Optional) A specific VLAN on the interface. More than one VLAN can be specified. The VLAN number that can be used is from 1 to 4094.

Command Default

All NDP messages are inspected. Secure Neighbor Discovery (SeND) options are ignored. Neighbors are probed based on the criteria defined in the Neighbor Tracking feature. Per-port IPv6 address limit enforcement is disabled. Layer 2 header source MAC address validations are disabled. Per-port rate limiting of the NDP messages in software is disabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(50)SY	This command was introduced.

Release	Modification
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SY.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 nd inspection** command applies the NDP Inspection feature on a specified interface. If you enable the optional **attach-policy** or **vlan** keywords, NDP traffic is inspected by policy or by VLAN. If no VLANs are specified, NDP traffic from all VLANs on the port is inspected (which is equivalent to using the **vlan all** keywords).

If no policy is specified in this command, the default criteria are as follows:

- All NDP messages are inspected.
- SeND options are ignored.
- Neighbors are probed based on the criteria defined in neighbor tracking feature.
- Per-port IPv6 address limit enforcement is disabled.
- Layer 2 header source MAC address validations are disabled.
- Per-port rate limiting of the NDP messages in software is disabled.

If a VLAN is specified, its parameter is either a single VLAN number from 1 to 4094 or a range of VLANs described by two VLAN numbers, the lesser one first, separated by a dash (for example, **vlan 1-100,200,300-400**). Do not enter any spaces between comma-separated VLAN parameters or in dash-specified ranges.

Examples

The following example enables NDP inspection on a specified interface:

```
Router(config-if)# ipv6 nd inspection
```

ipv6 nd inspection policy

To define the neighbor discovery (ND) inspection policy name and enter ND inspection policy configuration mode, use the **ipv6 nd inspection** command in ND inspection configuration mode. To remove the ND inspection policy, use the **no** form of this command.

ipv6 nd inspection policy *policy-name*

no ipv6 nd inspection policy *policy-name*

Syntax Description

policy-name	The ND inspection policy name.

Command Default

No ND inspection policies are configured.

Command Modes

ND inspection configuration (config-nd-inspection)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 nd inspection policy** command defines the ND inspection policy name and enters ND inspection policy configuration mode. Once you are in ND inspection policy configuration mode, you can use any of the following commands:

- **device-role**
- **drop-unsecure**
- **limit address-count**
- **sec-level minimum**
- **tracking**
- **trusted-port**
- **validate source-mac**

Examples

The following example defines an ND policy name as policy1:

```
Router(config)# ipv6 nd inspection policy policy1
Router(config-nd-inspection)#
```

Related Commands

Command	Description
device-role	Specifies the role of the device attached to the port.
drop-unsecure	Drops messages with no or invalid options or an invalid signature.
limit address-count	Limits the number of IPv6 addresses allowed to be used on the port.
sec-level minimum	Specifies the minimum security level parameter value when CGA options are used.
tracking	Overrides the default tracking policy on a port.
trusted-port	Configures a port to become a trusted port.
validate source-mac	Checks the source MAC address against the link-layer address.

ipv6 nd na glean

To configure neighbor discovery (ND) to glean an entry from an unsolicited neighbor advertisement (NA), use the **ipv6 nd na glean** command in interface configuration mode. To disable this feature, use the **no** form of this command.

ipv6 nd na glean

no ipv6 nd na glean

Syntax Description

This command has no arguments or keywords.

Command Default

The router ignores an unsolicited NA.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(33)SXI7	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

IPv6 nodes may choose to emit a multicast unsolicited NA packet following the successful completion of duplicate address detection (DAD). By default, these unsolicited NA packets are ignored by other IPv6 nodes. The **ipv6 nd na glean** command configures the router to create an ND entry on receipt of an unsolicited NA packet (assuming no such entry already exists and the NA has the link-layer address option). Use of this command allows a router to populate its ND cache with an entry for a neighbor in advance of any data traffic exchange with the neighbor.

Examples

The following example configures ND to glean an entry from an unsolicited neighbor advertisement:

```
Router(config-if)# ipv6 nd na glean
```

ipv6 nd nud retry

To configure the number of times neighbor unreachability detection (NUD) resends neighbor solicitations (NSs), use the **ipv6 nd nud retry** command in interface configuration mode. To disable this feature, use the **no** form of this command.

ipv6 nd nud retry *base interval max-attempts*

no ipv6 nd nud retry *base interval max-attempts*

Syntax Description

base	The base NUD value.
interval	The time interval, in milliseconds, between retries.
max-attempts	The maximum number of retry attempts, depending on the base value.

Command Default

Three NS packets are sent 1 second apart.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(33)SXI7	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

When a router runs NUD to re-resolve the ND entry for a neighbor, it sends three NS packets 1 second apart. In certain situations (for example, spanning-tree events, high traffic, the end host being reloaded), three NS packets sent at an interval of 1 second may not be sufficient. To help maintain the neighbor cache in such situations, use the **ipv6 nd nud retry** command to configure exponential timers for NS retransmits.

The maximum number of retry attempts is configured using the *max-attempts* argument. The retransmit interval is calculated with the following formula:

tm

- *t* = Time interval
- *m* = Base (1, 2, or 3)
- *n* = Current NS number (where the first NS is 0)

The **ipv6 nd nud retry** command affects only the retransmit rate for NUD, not for initial resolution, which uses the default of three NS packets sent 1 second apart.

Examples

The following example provides a fixed interval of 1 second and three retransmits:

```
Router(config-if)# ipv6 nd nud retry 1 1000 3
```

The following example provides a retransmit interval of 1, 2, 4, and 8:

```
Router(config-if)# ipv6 nd nud retry 2 1000 4
```

The following example provides the retransmit intervals of 1, 3, 9, 27, 81:

```
Router(config-if)# ipv6 nd nud retry 3 1000 5
```

ipv6 nd ra-throttle attach-policy

To attach an IPv6 router advertisement (RA) throttler policy to a Layer 2 interface or to a collection of VLANs, use the **ipv6 nd ra-throttle attach-policy** command in interface configuration mode or VLAN configuration mode. To remove the policy, use the **no** form of this command.

ipv6 nd ra-throttle attach-policy *policy-name*

Syntax Description

policy-name	RA throttler policy name.

Command Default

No policy is attached to an interface.

No policy is attached to a VLAN.

Command Modes

Interface configuration (config-if)

VLAN configuration (config-VLAN-config)

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

Use the **ipv6 nd ra-throttle attach-policy** command in interface configuration mode to attach the IPv6 RA throttler policy to a Layer 2 interface on the device port. Use the **ipv6 nd ra-throttle attach-policy** command in VLAN configuration mode to attach the IPv6 RA throttler policy to a VLAN or a collection of VLANs. To create the RA throttler policy, use the **ipv6 nd ra-throttle policy** command in global configuration mode.

IPv6 RA throttle policies must be attached at either the VLAN or BOX level in order to operate at the PORT level. If a policy or policies are attached at the PORT level only, IPv6 RA throttler will not function.

When a policy is applied on a port, any value that is not configured in the policy will be inherited from the VLAN configuration. If the value is not set in the VLAN configuration, then the default value is used.

Examples

The following example shows how to create an IPv6 RA throttler policy named policy1 and attach it to the Ethernet0/0 interface:

```
Device(config)# ipv6 nd ra-throttle policy policy1
Device(config-nd-ra-throttle)# exit
.
.
Device(config)# interface ethernet0/0
Device(config-if)# ipv6 nd ra-throttle attach-policy policy1
```

The following example shows how to create an IPv6 RA throttler policy named policy1 and attach it to a collection of VLANs named vlan1:

```
Device(config)# ipv6 nd ra-throttle policy policy1
Device(config-nd-ra-throttle)# exit
.
.
Device(config)# vlan configuration vlan1
Device(config-vlan-config)# ipv6 nd ra-throttle attach-policy policy1
```

ipv6 nd ra-throttle policy

To define the router advertisement (RA) throttler policy name and enter IPv6 RA throttle policy configuration mode, use the **ipv6 nd ra-throttle policy** command in global configuration mode. To reset the command to its defaults, use the **no** form of this command.

ipv6 nd ra-throttle policy *policy-name* **no ipv6 nd ra-throttle policy** *policy-name*

Syntax Description

policy-name	RA throttler policy name.

Command Default

- throttle-period: 600 seconds (10 minutes)
- max-through: 10 RAs per VLAN per 10 minutes.
- allow: at-least 1 at-most 1
- interval-option: passthrough
- medium-type: wired (port only)

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

Use the **ipv6 nd ra-throttle policy** command to define an IPv6 RA throttle policy and enter Pv6 RA throttle policy configuration mode.

The **allow at-least** and **allow at-most** command settings applied at the VLAN level provide the default for all devices in the VLAN. The values can be overwritten on a per-port basis by applying another policy on the specified port.

IPv6 RA throttle policies must be attached at either the VLAN or BOX level in order to operate at the PORT level. If a policy or policies are attached at the PORT level only, IPv6 RA throttler will not function.

When a policy is applied on a port, any value that is not configured in the policy will be inherited from the VLAN configuration. If the value is not set in the VLAN configuration, then the default value is used.

Examples

```
Device(config)# ipv6 nd ra-throttle policy policy1
Device(config-nd-ra-throttle)#
```

ipv6 nd raguard attach-policy

To apply the IPv6 router advertisement (RA) guard feature on a specified interface, use the **ipv6 nd raguard attach-policy** command in interface configuration mode.

ipv6 nd raguard attach-policy [*policy-name* [**vlan** {**add**| **except**| **none**| **remove**| **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]

Syntax Description

policy-name	(Optional) IPv6 RA guard policy name.
vlan	(Optional) Applies the IPv6 RA guard feature to a VLAN on the interface.
add	Adds a VLAN to be inspected.
except	All VLANs are inspected except the one specified.
none	No VLANs are inspected.
remove	Removes the specified VLAN from RA guard inspection.
all	ND traffic from all VLANs on the port is inspected.
vlan	(Optional) A specific VLAN on the interface. More than one VLAN can be specified (*vlan1*, *vlan2*, *vlan3*...). The range of available VLAN numbers is from 1 through 4094.

Command Default

An IPv6 RA guard policy is not configured.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

If no policy is specified using the *policy-name* argument, the port device role is set to host and all inbound router traffic (for example, RA and redirect messages) is blocked.

If no VLAN is specified (which is equal to entering the **vlan all** keywords after the *policy-name* argument), RA guard traffic from all VLANs on the port is analyzed.

If specified, the VLAN parameter is either a single VLAN number from 1 through 4094 or a range of VLANs described by two VLAN numbers, the lesser one first, separated by a dash. Do not enter any spaces between comma-separated vlan parameters or in dash-specified ranges; for example, vlan 1-100,200,300-400.

Examples

In the following example, the IPv6 RA guard feature is applied on GigabitEthernet interface 0/0:

```
Device(config)# interface GigabitEthernet 0/0
Device(config-if)# ipv6 nd raguard attach-policy
```

ipv6 nd raguard policy

To define the router advertisement (RA) guard policy name and enter RA guard policy configuration mode, use the **ipv6 nd raguard policy** command in global configuration mode.

ipv6 nd raguardpolicy *policy-name*

Syntax Description

policy-name	IPv6 RA guard policy name.

Command Default

An RA guard policy is not configured.

Command Modes

Global configuration (config)#

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Use the **ipv6 nd raguard policy** command to configure RA guard globally on a router. Once the device is in ND inspection policy configuration mode, you can use any of the following commands:

- **device-role**
- **drop-unsecure**
- **limit address-count**
- **sec-level minimum**
- **trusted-port**
- **validate source-mac**

After IPv6 RA guard is configured globally, you can use the **ipv6 nd raguard attach-policy** command to enable IPv6 RA guard on a specific interface.

Examples

The following example shows how to define the RA guard policy name as policy1 and place the device in policy configuration mode:

```
Device(config)# ipv6 nd raguard policy policy1
Device(config-ra-guard)#
```

Related Commands

Command	Description
device-role	Specifies the role of the device attached to the port.
drop-unsecure	Drops messages with no or invalid options or an invalid signature.
ipv6 nd raguard attach-policy	Applies the IPv6 RA guard feature on a specified interface.
limit address-count	Limits the number of IPv6 addresses allowed to be used on the port.
sec-level minimum	Specifies the minimum security level parameter value when CGA options are used.
trusted-port	Configures a port to become a trusted port.
validate source-mac	Checks the source MAC address against the link layer address.

ipv6 nd router-preference

To configure a default router preference (DRP) for the router on a specific interface, use the **ipv6 nd router-preference**command in interface configuration mode. To return to the default DRP, use the **no** form of this command.

ipv6 nd router-preference {**high**| **medium**| **low**}

no ipv6 nd router-preference

Syntax Description

high	Preference for the router specified on an interface is high.
medium	Preference for the router specified on an interface is medium.
low	Preference for the router specified on an interface is low.

Command Default

Router advertisements (RAs) are sent with the **medium** preference.

Command Modes

Interface configuration

Command History

Release	Modification
12.4(2)T	This command was introduced.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

RA messages are sent with the DRP configured by the **ipv6 nd router-preference** command. If no DRP is configured, RAs are sent with a medium preference.

A DRP is useful when, for example, two routers on a link may provide equivalent, but not equal-cost, routing, and policy may dictate that hosts should prefer one of the routers.

Examples

The following example configures a DRP of high for the router on gigabit Ethernet interface 0/1:

```
Router(config)# interface Gigabit ethernet 0/1
Router(config-if)# ipv6 nd router-preference high
```

Related Commands

Command	Description
ipv6 nd ra interval	Configures the interval between IPv6 router advertisement transmissions on an interface.
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.

ipv6 nd suppress attach-policy

To apply the IPv6 neighbor discovery (ND) suppress feature on a specified interface, use the **ipv6 nd suppress attach-policy** command in interface configuration mode.

ipv6 nd suppress attach-policy [*policy-name* [**vlan** {**add**| **except**| **none**| **remove**| **all**} *vlan* [*vlan1, vlan2, vlan3...*]]]

Syntax Description

policy-name	(Optional) IPv6 ND suppress policy name.
vlan	(Optional) Applies the IPv6 ND suppress feature to a VLAN on the interface.
add	Adds a VLAN to be inspected.
except	All VLANs are inspected except the one specified.
none	No VLANs are inspected.
remove	Removes the specified VLAN from IPv6 ND suppression.
all	ND traffic from all VLANs on the port is inspected.
vlan	(Optional) A specific VLAN on the interface. More than one VLAN can be specified (*vlan1*, *vlan2*, *vlan3*...). The range of available VLAN numbers is from 1 through 4094.

Command Default

An IPv6 ND suppress policy is not configured.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
15.3(1)S	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

If no VLAN is specified (which is equal to entering the **vlan all** keywords after the *policy-name* argument), RA guard traffic from all VLANs on the port is analyzed.

If specified, the VLAN parameter is either a single VLAN number from 1 through 4094 or a range of VLANs described by two VLAN numbers, the lesser one first, separated by a dash. Do not enter any spaces between comma-separated vlan parameters or in dash-specified ranges; for example, vlan 1-100,200,300-400.

Examples

In the following example, the IPv6 ND suppress feature is applied on Ethernet interface 0/0:

```
Device(config)# interface Ethernet 0/0
Device(config-if)# ipv6 nd suppress attach-policy
```

Related Commands

Command	Description
ipv6 nd suppress policy	Enables IPv6 ND multicast suppress and enter ND suppress policy configuration mode

ipv6 nd suppress policy

To enable IPv6 Neighbor Discovery (ND) multicast suppress and enter ND suppress policy configuration mode, use the **ipv6 nd suppress policy** command in global configuration mode.

ipv6 nd suppress policy *policy-name*

Syntax Description

policy-name	IPv6 ND suppress policy name.

Command Default

An ND suppress policy is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.3(1)S	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Use the **ipv6 nd suppress policy** command to configure NA suppress globally on a device. After IPv6 ND suppress is configured globally, you can use the **ipv6 nd suppress attach-policy** command to enable IPv6 ND suppress on a specific interface.

Examples

The following example shows how to define the ND suppress policy name as policy1 and place the device in policy configuration mode:

```
Device(config)# ipv6 nd suppress policy policy1
Device(config-nd-suppress)#
```

Related Commands

Command	Description
ipv6 nd suppress attach-policy	Applies the IPv6 ND suppress feature on a specified interface.

ipv6 neighbor binding logging

To enable the logging of binding table main events, use the **ipv6 neighbor binding logging** command in global configuration mode. To disable this function, use the **no** form of this command.

ipv6 neighbor binding logging

no ipv6 neighbor binding logging

Syntax Description

This command has no arguments or keywords.

Command Default

Binding table events are not logged.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 neighbor binding logging** command enables the logging of the following binding table events:

- An entry is inserted into the binding table.
- A binding table entry was updated.
- A binding table entry was deleted from the binding table.
- A binding table entry was not inserted into the binding table, possibly because of a collision with an existing entry, or because the maximum number of entries has been reached.

Examples

The following example shows how to enable binding table event logging:

```
Router(config)# ipv6 neighbor binding logging
```

Related Commands

Command	Description
ipv6 neighbor binding vlan	Adds a static entry to the binding table database.

Command	Description
ipv6 neighbor tracking	Tracks entries in the binding table.
ipv6 snooping logging packet drop	Configures IPv6 snooping security logging.

ipv6 neighbor binding max-entries

To specify the maximum number of entries that are allowed to be inserted in the binding table cache, use the **ipv6 neighbor binding max-entries** command in global configuration mode. To return to the default, use the **no** form of this command.

ipv6 neighbor binding max-entries *entries* [**vlan-limit** *number*| **interface-limit** *number*| **mac-limit** *number*]

no ipv6 neighbor binding max-entries *entries* [**vlan-limit**| **mac-limit**]

Syntax Description

entries	Number of entries that can be inserted into the cache.
vlan-limit *number*	(Optional) Specifies a neighbor binding limit per number of VLANs.
interface-limit *number*	(Optional) Specifies a neighbor binding limit per interface.
mac-limit *number*	(Optional) Specifies a neighbor binding limit per number of Media Access Control (MAC) addresses.

Command Default

This command is disabled.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 neighbor binding max-entries** command is used to control the content of the binding table. This command specifies the maximum number of entries that are allowed to be inserted in the binding table cache. Once this limit is reached, new entries are refused, and the Neighbor Discovery Protocol (NDP) traffic source with the new entry is dropped.

If the maximum number of entries specified is lower than the current number of entries in the database, no entries are cleared, and the new threshold is reached after normal cache attrition.

The maximum number of entries can be set globally by number of VLANs or by number of MAC addresses.

Examples

The following example shows how to specify globally the maximum number of entries inserted into the cache:

```
Router(config)# ipv6 neighbor binding max-entries 100
```

Related Commands

Command	Description
ipv6 neighbor binding vlan	Adds a static entry to the binding table database.
ipv6 neighbor tracking	Tracks entries in the binding table.

ipv6 neighbor binding vlan

To add a static entry to the binding table database, use the **ipv6 neighbor binding vlan** command in global configuration mode. To remove the static entry, use the **no** form of this command.

ipv6 neighbor binding vlan *vlan-id* {**interface** *type number*| *ipv6-address*| *mac-address*} [**tracking** [**disable**| **enable**| **retry-interval** *value*]| **reachable-lifetime** *value*]

no ipv6 neighbor binding vlan *vlan-id*

Syntax Description

vlan-id	ID of the specified VLAN.
interface *type number*	Adds static entries by the specified interface type and number.
ipv6-address	IPv6 address of the static entry.
mac-address	Media Access Control (MAC) address of the static entry.
tracking	(Optional) Verifies a static entry's reachability directly.
disable	(Optional) Disables tracking for a particular static entry.
enable	(Optional) Enables tracking for a particular static entry.
retry-interval *value*	(Optional) Verifies a static entry's reachability, in seconds, at the configured interval. The range is from 1 to 3600, and the default is 300.
reachable-lifetime *value*	(Optional) Specifies the maximum time, in seconds, an entry is considered reachable without getting a proof of reachability (direct reachability through tracking, or indirect reachability through Neighbor Discovery Protocol [NDP] inspection). After that, the entry is moved to stale. The range is from 1 to 3600 seconds, and the default is 300 seconds.

Command Default

Retry interval: 300 seconds

Reachable lifetime: 300 seconds

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 neighbor binding vlan** command is used to control the content of the binding table. Use this command to add a static entry in the binding table database. The binding table manager is responsible for aging out entries and verifying their reachability directly by probing them (if the **tracking** keyword is enabled). Use of the **tracking** keyword overrides any general behavior provided globally by the **ipv6 neighbor tracking** command for this static entry. The **disable** keyword disables tracking for this static entry. The **stale-lifetime** keyword defines the maximum time the entry will be kept once it is determined to be not reachable (or stale).

Examples

The following example shows how to change the reachable lifetime for binding entries to 100 seconds:

```
Router(config)# ipv6 neighbor binding vlan reachable-lifetime 100
```

Related Commands

Command	Description
ipv6 neighbor binding max-entries	Specifies the maximum number of entries that are allowed to be inserted in the cache.
ipv6 neighbor tracking	Tracks entries in the binding table.

ipv6 neighbor tracking

To track entries in the binding table, use the **ipv6 neighbor tracking** command in global configuration mode. To disable entry tracking, use the **no** form of this command.

ipv6 neighbor tracking [**retry-interval** *value*]

no ipv6 neighbor tracking [**retry-interval** *value*]

Syntax Description

retry-interval *value*	(Optional) Verifies a static entry's reachability at the configured interval time, in seconds, between two probings. The range is from 1 to 3600, and the default is 300.

Command Default

Retry interval: 300 seconds

Reachable lifetime: 300 seconds

Stale lifetime: 1440 minutes

Down lifetime: 1440 minutes

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **ipv6 neighbor tracking** command enables the tracking of entries in the binding table. Entry reachability is tested at every interval configured by the optional **retry-interval** keyword (or every 300 seconds, which is the default retry interval) using the neighbor unreachability detection (NUD) mechanism used for directly tracking neighbor reachability.

Reachability can also be established indirectly by using Neighbor Discovery Protocol (NDP) inspection up to the VERIFY_MAX_RETRIES value (the default is 10 seconds). When there is no response, entries are considered stale and are deleted after the stale lifetime value is reached (the default is 1440 minutes).

When the **ipv6 neighbor tracking** command is disabled, entries are considered stale after the reachable lifetime value is met (the default is 300 seconds) and deleted after the stale lifetime value is met.

To change the default values of neighbor binding entries in a binding table, use the **ipv6 neighbor binding** command.

Examples

The following example shows how to track entries in a binding table:

```
Router(config)# ipv6 neighbor tracking
```

Related Commands

Command	Description
ipv6 neighbor binding	Changes the defaults of neighbor binding entries in a binding table.

ipv6 prefix-list

To create an entry in an IPv6 prefix list, use the **ipv6 prefix-list** command in global configuration mode. To delete the entry, use the **no** form of this command.

ipv6 prefix-list *list-name* [**seq** *seq-number*] {**deny** *ipv6-prefix*/*prefix-length*| **permit** *ipv6-prefix*/*prefix-length*| **description** *text*} [**ge** *ge-value*] [**le** *le-value*]

no ipv6 prefix-list *list-name*

Syntax Description

list-name	Name of the prefix list. • Cannot be the same name as an existing access list. • Cannot be the name "detail" or "summary" because they are keywords in the **show ipv6 prefix-list** command.
seq *seq-number*	(Optional) Sequence number of the prefix list entry being configured.
deny	Denies networks that matches the condition.
permit	Permits networks that matches the condition.
ipv6-prefix	The IPv6 network assigned to the specified prefix list. 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 comprise the prefix (the network portion of the address). A slash mark must precede the decimal value.
description *text*	A description of the prefix list that can be up to 80 characters in length.
ge *ge-value*	(Optional) Specifies a prefix length greater than or equal to the *ipv6-prefix*/*prefix-length* arguments. It is the lowest value of a range of the *length* (the "from" portion of the length range).

le *le-value*	(Optional) Specifies a prefix length less than or equal to the *ipv6-prefix /prefix-length* arguments. It is the highest value of a range of the *length* (the “to” portion of the length range).

Command Default

No prefix list is created.

Command Modes

Global configuration (config)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

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

To suppress networks from being advertised in updates, use the **distribute-list out** command.

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. For efficiency, you may want to put the most common permits or denies near the top of the list, using the *seq-number* argument.

The **show ipv6 prefix-list** command displays the sequence numbers of entries.

IPv6 prefix lists are used to specify certain prefixes or a range of prefixes that must be matched before a permit or deny statement can be applied. Two operand keywords can be used to designate a range of prefix lengths to be matched. A prefix length of less than, or equal to, a value is configured with the **le** keyword. A prefix length greater than, or equal to, a value is specified using the **ge** keyword. The **ge** and **le** keywords can be used to specify the range of the prefix length to be matched in more detail than the usual *ipv6-prefix*/*prefix-length* argument. For a candidate prefix to match against a prefix list entry three conditions can exist:

- The candidate prefix must match the specified prefix list and prefix length entry.
- The value of the optional **le** keyword specifies the range of allowed prefix lengths from the *prefix-length* argument up to, and including, the value of the **le** keyword.
- The value of the optional **ge** keyword specifies the range of allowed prefix lengths from the value of the **ge** keyword up to, and including, 128.

Note The first condition must match before the other conditions take effect.

An exact match is assumed when the **ge** or **le** keywords are not specified. If only one keyword operand is specified then the condition for that keyword is applied, and the other condition is not applied. The *prefix-length* value must be less than the **ge** value. The **ge** value must be less than, or equal to, the **le** value. The **le** value must be less than or equal to 128.

Every IPv6 prefix list, including prefix lists that do not have any permit and deny condition statements, has an implicit deny any any statement as its last match condition.

Examples

The following example denies all routes with a prefix of ::/0.

```
Router(config)# ipv6 prefix-list abc deny ::/0
```

The following example permits the prefix 2002::/16:

```
Router(config)# ipv6 prefix-list abc permit 2002::/16
```

The following example shows how to specify a group of prefixes to accept any prefixes from prefix 5F00::/48 up to and including prefix 5F00::/64.

```
Router(config)# ipv6 prefix-list abc permit 5F00::/48 le 64
```

The following example denies prefix lengths greater than 64 bits in routes that have the prefix 2001:0DB8::/64.

```
Router(config)# ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
```

The following example permits mask lengths from 32 to 64 bits in all address space.

```
Router(config)# ipv6 prefix-list abc permit ::/0 ge 32 le 64
```

The following example denies mask lengths greater than 32 bits in all address space.

```
Router(config)# ipv6 prefix-list abc deny ::/0 ge 32
```

The following example denies all routes with a prefix of 2002::/128.

```
Router(config)# ipv6 prefix-list abc deny 2002::/128
```

The following example permits all routes with a prefix of ::/0.

```
Router(config)# ipv6 prefix-list abc permit ::/0
```

Related Commands

Command	Description
clear ipv6 prefix-list	Resets the hit count of the IPv6 prefix list entries.
distribute-list out	Suppresses networks from being advertised in updates.
ipv6 prefix-list sequence-number	Enables the generation of sequence numbers for entries in an IPv6 prefix list.
match ipv6 address	Distributes IPv6 routes that have a prefix permitted by a prefix list.
show ipv6 prefix-list	Displays information about an IPv6 prefix list or IPv6 prefix list entries.

ipv6-i4

ipv6 snooping attach-policy

To apply an IPv6 snooping policy to a target, use the **ipv6 snooping attach-policy** command in IPv6 snooping configuration mode. To remove a policy from a target, use the **no** form of this command.

ipv6 snooping policy attach-policy *snooping-policy*

Syntax Description

snooping-policy	User-defined name of the snooping policy. The policy name can be a symbolic string (such as Engineering) or an integer (such as 0).

Command Default

An IPv6 snooping policy is not attached to a target.

Command Modes

IPv6 snooping configuration (config-ipv6-snooping)

Command History

Release	Modification
15.0(2)SE	This command was introduced.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Once a policy has been identified or configured, it is applied on a target using the **ipv6 snooping attach-policy** command. This command is applied on any target, which varies depending on the platform. Examples of targets (depending on the platform used) include device ports, switchports, Layer 2 interfaces, Layer 3 interfaces, and VLANs.

Examples

The following examples shows how to apply an IPv6 snooping policy named policy1 to a target:

```
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# ipv6 snooping attach-policy policy1
```

Related Commands

Command	Description
ipv6 snooping policy	Configures an IPv6 snooping policy and enters IPv6 snooping configuration mode.

ipv6 snooping policy

To configure an IPv6 snooping policy and enter IPv6 snooping configuration mode, use the **ipv6 snooping policy** command in global configuration mode. To delete an IPv6 snooping policy, use the **no** form of this command.

ipv6 snooping policy *snooping-policy*

no ipv6 snooping policy *snooping-policy*

Syntax Description

snooping-policy	User-defined name of the snooping policy. The policy name can be a symbolic string (such as Engineering) or an integer (such as 0).

Command Default

An IPv6 snooping policy is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
15.0(2)SE	This command was introduced.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Use the **ipv6 snooping policy** command to create an IPv6 snooping policy. When the **ipv6 snooping policy** command is enabled, the configuration mode changes to IPv6 snooping configuration mode. In this mode, the administrator can configure the following IPv6 first-hop security commands:

- The **data-glean**/**destination-glean** command enables IPv6 first-hop security binding table recovery using data or destination address gleaning.
- The **device-role** command specifies the role of the device attached to the port.
- The **limit address-count** *maximum* command limits the number of IPv6 addresses allowed to be used on the port.
- **security-level** specifies the level of security enforced.
- The **tracking** command overrides the default tracking policy on a port.

- The **trusted-port** command configures a port to become a trusted port; that is, limited or no verification is performed when messages are received.

Once a policy has been identified or configured, it is applied on a device using the **ipv6 snooping attach-policy** command.

Examples

The following examples show hows to configure an IPv6 snooping policy:

```
Device(config)# ipv6 snooping policy policy1
```

Related Commands

Command	Description
ipv6 snooping attach-policy	Applies an IPv6 snooping policy to a target.

ipv6 traffic-filter

To filter incoming or outgoing IPv6 traffic on an interface, use the **ipv6 traffic-filter**command in interface configuration mode. To disable the filtering of IPv6 traffic on an interface, use the **no** form of this command.

ipv6 traffic-filter *access-list-name* {**in**| **out**}

no ipv6 traffic-filter *access-list-name*

Syntax Description

access-list-name	Specifies an IPv6 access name.
in	Specifies incoming IPv6 traffic.
out	Specifies outgoing IPv6 traffic.

Command Default

Filtering of IPv6 traffic on an interface is not configured.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 series routers.
12.2(33)SXI4	The **out** keyword and therefore filtering of outgoing traffic is not supported in IPv6 port-based access list (PACL) configuration.
12.2(54)SG	This command was modified. Support for Cisco IOS Release 12.2(54)SG was added.

Release	Modification
12.2(50)SY	This command was modified. The **out** keyword is not supported.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Examples

The following example filters inbound IPv6 traffic on Ethernet interface 0/0 as defined by the access list named cisco:

```
Router(config)# interface ethernet 0/0
Router(config-if)# ipv6 traffic-filter cisco in
```

Related Commands

Command	Description
ipv6 access-list	Defines an IPv6 access list and sets deny or permit conditions for the defined access list.
show ipv6 access-list	Displays the contents of all current IPv6 access lists.
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.

ipv6 verify unicast source reachable-via

To verify that a source address exists in the FIB table and enable Unicast Reverse Path Forwarding (Unicast RPF), use the **ipv6 verify unicast source reachable-via** command in interface configuration mode. To disable URPF, use the **no** form of this command.

ipv6 verify unicast source reachable-via {**rx**| **any**} **[allow-default] [allow-self-ping]** [*access-list-name*]

no ipv6 verify unicast

Syntax Description

rx	Source is reachable through the interface on which the packet was received.
any	Source is reachable through any interface.
allow-default	(Optional) Allows the lookup table to match the default route and use the route for verification.
allow-self-ping	(Optional) Allows the router to ping a secondary address.
access-list-name	(Optional) Name of the IPv6 access list. Names cannot contain a space or quotation mark, or begin with a numeral.

Command Default

Unicast RPF is disabled.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(25)S	This command was introduced.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Series Aggregation Services Routers.

Usage Guidelines

The **ipv6 verify unicast reverse-path** command is used to enable Unicast RPF for IPv6 in loose checking mode.

Use the **ipv6 verify unicast source reachable-via**command to mitigate problems caused by malformed or forged (spoofed) IP source addresses that pass through an IPv6 router. Malformed or forged source addresses can indicate denial-of-service (DoS) attacks based on source IPv6 address spoofing.

The URPF feature checks to see if any packet received at a router interface arrives on one of the best return paths to the source of the packet. The feature does this by doing a reverse lookup in the CEF table. If URPF does not find a reverse path for the packet, U RPF can drop or forward the packet, depending on whether an access control list (ACL) is specified in the **ipv6 verify unicast source reachable-via** command. If an ACL is specified in the command, then when (and only when) a packet fails the URPF check, the ACL is checked to see if the packet should be dropped (using a deny statement in the ACL) or forwarded (using a permit statement in the ACL). Whether a packet is dropped or forwarded, the packet is counted in the global IP traffic statistics for U RPF drops and in the interface statistics for Unicast RPF.

If no ACL is specified in the **ipv6 verify unicast source reachable-via** command, the router drops the forged or malformed packet immediately and no ACL logging occurs. The router and interface Unicast RPF counters are updated.

U RPF events can be logged by specifying the logging option for the ACL entries used by the **ipv6 verify unicast source reachable-via** command. Log information can be used to gather information about the attack, such as source address, time, and so on.

Examples

The following example enables Unicast RPF on any interface:

```
ipv6 verify unicast source reachable-via any
```

Related Commands

Command	Description
ipv6 access-list	Defines an IPv6 access list and places the router in IPv6 access list configuration mode.
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.

managed-config-flag

To verify the advertised managed address configuration parameter, use the **managed-config-flag** command in RA guard policy configuration mode.

managed-config-flag {**on**| **off**}

Syntax Description

on	Verification is enabled.
off	Verification is disabled.

Command Default

Verification is not enabled.

Command Modes

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **managed-config-flag** command enables verification of the advertised managed address configuration parameter (or "M" flag). This flag could be set by an attacker to force hosts to obtain addresses through a DHCPv6 server that may not be trustworthy.

Examples

The following example shows how the command defines a router advertisement (RA) guard policy name as raguard1, places the router in RA guard policy configuration mode, and enables M flag verification:

```
Router(config)# ipv6 nd raguard policy raguard1
Router(config-ra-guard)# managed-config-flag on
```

Related Commands

Command	Description
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.

match ipv6

To configure one or more of the IPv6 fields as a key field for a flow record, use the **match ipv6** command in Flexible NetFlow flow record configuration mode. To disable the use of one or more of the IPv6 fields as a key field for a flow record, use the **no** form of this command.

match ipv6 {**dscp**| **flow-label**| **next-header**| **payload-length**| **precedence**| **protocol**| **traffic-class**| **version**}

no match ipv6 {**dscp**| **flow-label**| **next-header**| **payload-length**| **precedence**| **protocol**| **traffic-class**| **version**}

Cisco Catalyst 6500 Switches in Cisco IOS Release 12.2(50)SY

match ipv6 {**dscp**| **precedence**| **protocol**| **tos**}

no match ipv6 {**dscp**| **precedence**| **protocol**| **tos**}

Cisco IOS XE Release 3.2SE

match ipv6 {**protocol**| **traffic-class**| **version**}

no match ipv6 {**protocol**| **traffic-class**| **version**}

Syntax Description

dscp	Configures the IPv6 differentiated services code point DSCP (part of type of service (ToS)) as a key field.
flow-label	Configures the IPv6 flow label as a key field.
next-header	Configures the IPv6 next header as a key field.
payload-length	Configures the IPv6 payload length as a key field.
precedence	Configures the IPv6 precedence (part of ToS) as a key field.
protocol	Configures the IPv6 protocol as a key field.
tos	Configures the IPv6 ToS as a key field.
traffic-class	Configures the IPv6 traffic class as a key field.
version	Configures the IPv6 version from IPv6 header as a key field.

Command Default

The IPv6 fields are not configured as a key field.

Command Modes

Flexible Netflow flow record configuration (config-flow-record)

Command History

Release	Modification
12.4(20)T	This command was introduced.
12.2(33)SRE	This command was modified. Support for this command was implemented on the Cisco 7200 and Cisco 7300 Network Processing Engine (NPE) series routers.
12.2(50)SY	This command was modified. The **flow-label**, **next-header**, **payload-length,traffic-class**, and **version** keywords were removed.
15.2(2)T	This command was modified. Support for the Cisco Performance Monitor was added.
Cisco IOS XE Release 3.5S	This command was modified. Support for the Cisco Performance Monitor was added.
Cisco IOS XE Release 3.2SE	This command was modified. The **dscp**, **flow-label**, **next-header**, **payload-length**, and **precedence** keywords were removed.

Usage Guidelines

This command can be used with both Flexible NetFlow and Performance Monitor. These products use different commands to enter the configuration mode in which you issue this command, however the mode prompt is the same for both products. For Performance Monitor, you must first enter the **flow record type performance-monitor** command before you can use this command.

Because the mode prompt is the same for both products, here we refer to the command mode for both products as flow record configuration mode. However, for Flexible NetFlow, the mode is also known as Flexible NetFlow flow record configuration mode; and for Performance Monitor, the mode is also known as Performance Monitor flow record configuration mode.

A flow record requires at least one key field before it can be used in a flow monitor. The key fields differentiate flows, with each flow having a unique set of values for the key fields. The key fields are defined using the **match** command.

Note Some of the keywords of the **match ipv6** command are documented as separate commands. All of the keywords for the **match ipv6** command that are documented separately start with **match ipv6**. For example, for information about configuring the IPv6 hop limit as a key field for a flow record, refer to the **match ipv6 hop-limit** command.

Examples

The following example configures the IPv6 DSCP field as a key field:

```
Router(config)# flow record FLOW-RECORD-1
Router(config-flow-record)# match ipv6 dscp
```

The following example configures the IPv6 DSCP field as a key field:

```
Router(config)# flow record type performance-monitor RECORD-1
Router(config-flow-record)# match ipv6 dscp
```

Related Commands

Command	Description
flow record	Creates a flow record, and enters Flexible NetFlow flow record configuration mode.
flow record type performance-monitor	Creates a flow record, and enters Performance Monitor flow record configuration mode.

match ipv6 access-list

To verify the sender’s IPv6 address in inspected messages from the authorized prefix list, use the **match ipv6 access-list** command in RA guard policy configuration mode.

match ipv6 access-list *ipv6-access-list-name*

Syntax Description

ipv6-access-list-name	The IPv6 access list to be matched.

Command Default

Senders’ IPv6 addresses are not verified.

Command Modes

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **match ipv6 access-list** command enables verification of the sender’s IPv6 address in inspected messages from the configured authorized router source access list. If the **match ipv6 access-list** command is not configured, this authorization is bypassed.

An access list is configured using the **ipv6 access-list** command. For instance, to authorize the router with link-local address FE80::A8BB:CCFF:FE01:F700 only, define the following IPv6 access list:

```
Router(config)# ipv6 access-list list1
Router(config-ipv6-acl)# permit host FE80::A8BB:CCFF:FE01:F700 any
```

Note The access list is used here as a convenient way to define several explicit router sources, but it should not be considered to be a port-based access list (PACL). The **match ipv6 access-list** command verifies the IPv6 source address of the router messages, so specifying a destination in the access list is meaningless and the destination of the access control list (ACL) entry should always be "any." If a destination is specified in the access list, then matching will fail.

Examples

The following example shows how the command defines a router advertisement (RA) guard policy name as raguard1, places the router in RA guard policy configuration mode, and matches the IPv6 addresses in the access list named list1:

```
Router(config)# ipv6 nd raguard policy raguard1
Router(config-ra-guard)# match ipv6 access-list list1
```

Related Commands

Command	Description
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.
ipv6 access-list	Defines an IPv6 access list and places the router in IPv6 access list configuration mode.

match ipv6 address

To distribute IPv6 routes that have a prefix permitted by a prefix list or to specify an IPv6 access list to be used to match packets for policy-based routing (PBR) for IPv6, use the **match ipv6 address** command in route-map configuration mode. To remove the **match ipv6 address** entry, use the **no** form of this command.

match ipv6 address {**prefix-list** *prefix-list-name*| *access-list-name*}

no match ipv6 address

Syntax Description

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

Command Default

No routes are distributed based on the destination network number or an access list.

Command Modes

Route-map configuration (config-route-map)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.3(7)T	This command was modified. The *access-list-name* argument was added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SXI4	This command was modified. The **prefix-list** *prefix-list-name* keyword-argument pair argument is not supported in Cisco IOS Release 12.2(33)SXI4.
Cisco IOS XE Release 3.2S	This command was integrated into Cisco IOS XE Release 3.2S.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Use the **route-map** command and the **match** and **set** commands to define the conditions for redistributing routes from one routing protocol to another. Each **route-map** command has a list of **match** and **set** commands associated with it. The **match** commands specify the match criteria--the conditions under which redistribution is allowed for the current **route-map** command. The **set** commands specify the set actions, which are the particular redistribution actions to be performed if the criteria enforced by the **match** commands are met.

The **match ipv6 address** command can be used to specify either an access list or a prefix list. When using PBR, you must use the *access-list-name* argument; the **prefix-list** *prefix-list-name* keyword-argument pair argument will not work.

Examples

In the following example, IPv6 routes that have addresses specified by the prefix list named marketing are matched:

```
Device(config)# route-map name
Device(config-route-map)# match ipv6 address prefix-list marketing
```

In the following example, IPv6 routes that have addresses specified by an access list named marketing are matched:

```
Device(config)# route-map
Device(config-route-map)# match ipv6 address marketing
```

Related Commands

Command	Description
match as-path	Matches a BGP autonomous system path access list.
match community	Matches a BGP community.
match ipv6 address	Specifies an IPv6 access list to be used to match packets for PBR for IPv6.
match ipv6 next-hop	Distributes IPv6 routes that have a next-hop prefix permitted by a prefix list.
match ipv6 route-source	Distributes IPv6 routes that have been advertised by routers at an address specified by a prefix list.
match length	Bases policy routing on the Level 3 length of a packet.
match metric	Redistributes routes with the specified metric.
match route-type	Redistributes routes of the specified type.
route-map	Defines conditions for redistributing routes from one routing protocol into another.
set as-path	Modifies an autonomous system path for BGP routes.
set community	Sets the BGP community attribute.

Command	Description
set default interface	Specifies the default interface to output packets that pass a match clause of a route map for policy routing and have no explicit route to the destination.
set interface	Specifies the default interface to output packets that pass a match clause of a route map for policy routing.
set ipv6 default next-hop	Specifies an IPv6 default next hop to which matching packets will be forwarded.
set ipv6 next-hop (PBR)	Indicates where to output IPv6 packets that pass a match clause of a route map for policy routing.
set ipv6 precedence	Sets the precedence value in the IPv6 packet header.
set level	Indicates where to import routes.
set local preference	Specifies a preference value for the autonomous system path.
set metric	Sets the metric value for a routing protocol.
set metric-type	Sets the metric type for the destination routing protocol.
set tag	Sets a tag value of the destination routing protocol.
set weight	Specifies the BGP weight for the routing table.

match ipv6 destination

To configure the IPv6 destination address as a key field for a flow record, use the **match ipv6 destination** command in Flexible Netflow flow record configuration mode. To disable the IPv6 destination address as a key field for a flow record, use the **no** form of this command.

match ipv6 destination {**address**| {**mask**| **prefix**} [**minimum-mask** *mask*]}

no match ipv6 destination {**address**| {**mask**| **prefix**} [**minimum-mask** *mask*]}

Cisco Catalyst 6500 Switches in Cisco IOS Release 12.2(50)SY

match ipv6 destination address

no match ipv6 destination address

Cisco IOS XE Release 3.2SE

match ipv6 destination address

no match ipv6 destination address

Syntax Description

address	Configures the IPv6 destination address as a key field.
mask	Configures the mask for the IPv6 destination address as a key field.
prefix	Configures the prefix for the IPv6 destination address as a key field.
minimum-mask *mask*	(Optional) Specifies the size, in bits, of the minimum mask. Range: 1 to 128.

Command Default

The IPv6 destination address is not configured as a key field.

Command Modes

Flexible NetFlow flow record configuration (config-flow-record)

Command History

Release	Modification
12.4(20)T	This command was introduced.
12.2(33)SRE	This command was modified. Support for this command was implemented on the Cisco 7200 and Cisco 7300 Network Processing Engine (NPE) series routers.

Release	Modification
12.2(50)SY	This command was modified. The **mask**, **prefix**, and **minimum-mask** keywords were removed.
15.2(2)T	This command was modified. Support for the Cisco Performance Monitor was added.
Cisco IOS XE Release 3.5S	This command was modified. Support for the Cisco Performance Monitor was added.
Cisco IOS XE Release 3.2SE	This command was modified. The **mask**, **prefix**, and **minimum-mask** keywords were removed.

Usage Guidelines

This command can be used with both Flexible NetFlow and Performance Monitor. These products use different commands to enter the configuration mode in which you issue this command, however the mode prompt is the same for both products. For Performance Monitor, you must first enter the **flow record type performance-monitor** command before you can use this command.

Because the mode prompt is the same for both products, here we refer to the command mode for both products as flow record configuration mode. However, for Flexible NetFlow, the mode is also known as Flexible NetFlow flow record configuration mode; and for Performance Monitor, the mode is also known as Performance Monitor flow record configuration mode.

A flow record requires at least one key field before it can be used in a flow monitor. The key fields differentiate flows, with each flow having a unique set of values for the key fields. The key fields are defined using the **match** command.

Examples

The following example configures a 16-bit IPv6 destination address prefix as a key field:

```
Router(config)# flow record FLOW-RECORD-1
Router(config-flow-record)# match ipv6 destination prefix minimum-mask 16
```

The following example specifies a 16-bit IPv6 destination address mask as a key field:

```
Router(config)# flow record FLOW-RECORD-1
Router(config-flow-record)# match ipv6 destination mask minimum-mask 16
```

The following example configures a 16-bit IPv6 destination address mask as a key field:

```
Router(config)# flow record type performance-monitor RECORD-1
Router(config-flow-record)# match ipv6 destination mask minimum-mask 16
```

Related Commands

Command	Description
flow record	Creates a flow record, and enters Flexible NetFlow flow record configuration mode.
flow record type performance-monitor	Creates a flow record, and enters Performance Monitor flow record configuration mode.

match ipv6 hop-limit

To configure the IPv6 hop limit as a key field for a flow record, use the **match ipv6 hop-limit** command in Flexible NetFlow flow record configuration mode. To disable the use of a section of an IPv6 packet as a key field for a flow record, use the **no** form of this command.

match ipv6 hop-limit

no match ipv6 hop-limit

Syntax Description

This command has no arguments or keywords.

Command Default

The use of the IPv6 hop limit as a key field for a user-defined flow record is not enabled by default.

Command Modes

Flexible NetFlow flow record configuration (config-flow-record)

Command History

Release	Modification
12.4(20)T	This command was introduced.
12.2(33)SRE	This command was modified. Support for this command was implemented on the Cisco 7200 and Cisco 7300 Network Processing Engine (NPE) series routers.
15.2(2)T	This command was modified. Support for the Cisco Performance Monitor was added.
Cisco IOS XE Release 3.5S	This command was modified. Support for the Cisco Performance Monitor was added.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

This command can be used with both Flexible NetFlow and Performance Monitor. These products use different commands to enter the configuration mode in which you issue this command, however the mode prompt is the same for both products. For Performance Monitor, you must first enter the **flow record type performance-monitor** command before you can use this command.

Because the mode prompt is the same for both products, here we refer to the command mode for both products as flow record configuration mode. However, for Flexible NetFlow, the mode is also known as Flexible NetFlow flow record configuration mode; and for Performance Monitor, the mode is also known as Performance Monitor flow record configuration mode.

A flow record requires at least one key field before it can be used in a flow monitor. The key fields differentiate flows, with each flow having a unique set of values for the key fields. The key fields are defined using the **match** command.

Examples

The following example configures the hop limit of the packets in the flow as a key field:

```
Router(config)# flow record FLOW-RECORD-1
Router(config-flow-record)# match ipv6 hop-limit
```

The following example configures the hop limit of the packets in the flow as a key field:

```
Router(config)# flow record type performance-monitor RECORD-1
Router(config-flow-record)# match ipv6 hop-limit
```

Related Commands

Command	Description
flow record	Creates a flow record, and enters Flexible NetFlow flow record configuration mode.
flow record type performance-monitor	Creates a flow record, and enters Performance Monitor flow record configuration mode.

match ra prefix-list

To verify the advertised prefixes in inspected messages from the authorized prefix list, use the **match ra prefix-list** command in RA guard policy configuration mode.

match ra prefix-list *ipv6-prefix-list-name*

Syntax Description

ipv6-prefix-list-name	The IPv6 prefix list to be matched.

Command Default

Advertised prefixes are not verified.

Command Modes

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **match ra prefix-list** command enables verification of the advertised prefixes in inspected messages from the configured authorized prefix list. Use the **ipv6 prefix-list** command to configure an IPv6 prefix list. For instance, to authorize the 2001:101::/64 prefixes and deny the 2001:100::/64 prefixes, define the following IPv6 prefix list:

```
Router(config)# ipv6 prefix-list listname1 deny 2001:0DB8:101:/64
Router(config)# ipv6 prefix-list listname1 permit 2001:0DB8:100::/64
```

Examples

The following example shows how the command defines an router advertisement (RA) guard policy name as raguard1, places the router in RA guard policy configuration mode, and verifies the advertised prefixes in listname1:

```
Router(config)# ipv6 nd raguard policy raguard1
Router(config-ra-guard)# match ra prefix-list listname1
```

Related Commands

Command	Description
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.
ipv6 prefix-list	Creates an entry in an IPv6 prefix list.

max-through

To limit multicast router advertisements (RAs) per VLAN per throttle period, use the **max-through** command in IPv6 RA throttle policy configuration mode. To reset the command to its defaults, use the **no** form of this command.

max-through {*mt-value*| **inherit**| **no-limit**}

Syntax Description

mt-value	Number of multicast RAs allowed on the VLAN before throttling occurs. The range is from 0 through 256.
inherit	Merges the setting between target policies.
no-limit	Multicast RAs are not limited on the VLAN.

Command Default

10 RAs per VLAN per 10 minutes

Command Modes

IPv6 RA throttle policy configuration (config-nd-ra-throttle)

Command History

Release	Modification
Cisco IOS XE Release 3.2XE	This command was introduced.

Usage Guidelines

The **max-through** command limits the amount of multicast RAs that are passed through to the VLAN per throttle period. This command can be configured only on a VLAN.

Examples

```
Device(config)# ipv6 nd ra-throttle policy policy1
Device(config-nd-ra-throttle)# max-through 25
```

medium-type

To indicate whether a device is wired or wireless, use the **medium-type** command in IPv6 RA throttle policy configuration mode. To reset the command to its defaults, use the **no** form of this command.

medium-type {**access-point**| **wired**}

Syntax Description

access-point	The attached device is a radio access point and is throttled.
wired	The attached device is wired and is not throttled.

Command Default

Wired

Command Modes

IPv6 RA throttle policy configuration (config-nd-ra-throttle)

Command History

Release	Modification
Cisco IOS XE Release XE3.2S	This command was introduced.

Usage Guidelines

The **medium-type** command indicates the type of access on a port only. The VLAN ignores any values specified by the **medium-type** command.

Examples

```
Device(config)# ipv6 nd ra-throttle policy policy1
Device(config-nd-ra-throttle)# medium-type wired
```

mode dad-proxy

To enable duplicate address detection (DAD) proxy mode for IPv6 Neighbor Discovery (ND) suppress, use the **mode dad-proxy** command in ND suppress policy configuration mode. To disable this feature, use the **no** form of this command.

mode dad-proxy

Syntax Description

This command has no arguments or keywords.

Command Default

All multicast neighbor solicitation (NS) messages are suppressed.

Command Modes

ND suppress policy configuration mode (config-nd-suppress)

Command History

Release	Modification
15.1(2)SG	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The IPv6 Dad proxy feature responds on behalf of the address's owner when an address is already in use. Use the **mode dad-proxy** command to enable IPv6 DAD proxy when using IPv6 ND suppress. If your device does not support IPv6 multicast suppress, you can enable IPv6 DAD proxy by entering the **ipv6 nd dad-proxy** command in global configuration mode.

Examples

```
Device(config)# ipv6 nd suppress policy policy1
Device(config-nd-suppress)# mode dad-proxy
```

Related Commands

Command	Description
ipv6 nd dad-proxy	Enables the IPv6 ND DAD proxy feature on the device.
ipv6 nd suppress policy	Enables IPv6 ND multicast suppress and enters ND suppress policy configuration mode.

network (IPv6)

To configure the network source of the next hop to be used by the PE VPN, use the network command in router configuration mode. To disable the source, use the **no** form of this command.

network *ipv6-address/prefix-length*

no network *ipv6-address/prefix-length*

Syntax Description

ipv6-address	The 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.

Command Default

Next-hop network sources are not configured.

Command Modes

Address family configuration Router configuration

Command History

Release	Modification
12.2(33)SRB	This command was introduced.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.
12.2(33)SXI	This command was integrated into Cisco IOS Release 12.2(33)SXI.
Cisco IOS XE Release 3.1S	This command was integrated into Cisco IOS XE Release 3.1S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The *ipv6-address* argument in this command configures the IPv6 network number.

Examples

The following example places the router in address family configuration mode and configures the network source to be used as the next hop:

```
Router(config)# router bgp 100
Router(config-router)# network 2001:DB8:100::1/128
```

Related Commands

Command	Description
address-family ipv6	Enters address family configuration mode for configuring routing sessions such as BGP that use standard IPv6 address prefixes.
address-family vpnv6	Places the router in address family configuration mode for configuring routing sessions that use standard VPNv6 address prefixes.

other-config-flag

To verify the advertised “other” configuration parameter, use the **other-config-flag** command in RA guard policy configuration mode.

other-config-flag {**on**| **off**}

Syntax Description

on	Verification is enabled.
off	Verification is disabled.

Command Default

Verification is not enabled.

Command Modes

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **other-config-flag** command enables verification of the advertised "other" configuration parameter (or "O" flag). This flag could be set by an attacker to force hosts to retrieve other configuration information through a Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server that may not be trustworthy.

Examples

The following example shows how the command defines a router advertisement (RA) guard policy name as raguard1, places the router in RA guard policy configuration mode, and enables O flag verification:

```
Router(config)# ipv6 nd raguard policy raguard1
Router(config-ra-guard)# other-config-flag on
```

Related Commands

Command	Description
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.

passive-interface (IPv6)

To disable sending routing updates on an interface, use the **passive-interface** command in router configuration mode. To reenable the sending of routing updates, use the **no** form of this command.

passive-interface [**default**| *interface-type interface-number*]

no passive-interface [**default**| *interface-type interface-number*]

Syntax Description

default	(Optional) All interfaces become passive.
interface-type interface-number	(Optional) Interface type and number. For more information, use the question mark (**?**) online help function.

Command Default

No interfaces are passive. Routing updates are sent to all interfaces on which the routing protocol is enabled.

Command Modes

Router configuration

Command History

Release	Modification
12.2(15)T	This command was introduced.
12.4(6)T	Support for Enhanced Internal Gateway Routing Protocol (EIGRP) IPv6 was added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRB	This command was integrated into Cisco IOS Release 12.2(33)SRB.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

If you disable the sending of routing updates on an interface, the particular address prefix will continue to be advertised to other interfaces, and updates from other routers on that interface continue to be received and processed.

The **default** keyword sets all interfaces as passive by default. You can then configure individual interfaces where adjacencies are desired using the **no passive-interface** command. The **default** keyword is useful in Internet service provider (ISP) and large enterprise networks where many of the distribution routers have more than 200 interfaces.

OSPF for IPv6 routing information is neither sent nor received through the specified router interface. The specified interface address appears as a stub network in the OSPF for IPv6 domain.

For the Intermediate System-to-Intermediate System (IS-IS) protocol, this command instructs IS-IS to advertise the IP addresses for the specified interface without actually running IS-IS on that interface. The **no** form of this command for IS-IS disables advertising IP addresses for the specified address.

Examples

The following example sets all interfaces as passive, then activates Ethernet interface 0:

```
Router(config-router)# passive-interface default
Router(config-router)# no passive-interface ethernet0/0
```

passive-interface (OSPFv3)

To suppress sending routing updates on an interface when using an IPv4 Open Shortest Path First version 3 (OSPFv3) process, use the **passive-interface** command in router configuration mode. To reenable the sending of routing updates, use the **no** form of this command.

passive-interface [**default**| *interface-type interface-number*]

no passive-interface [**default**| *interface-type interface-number*]

Syntax Description

default	(Optional) All interfaces become passive.
interface-type interface-number	(Optional) Interface type and number. For more information, use the question mark (**?**) online help function.

Command Default

No interfaces are passive. Routing updates are sent to all interfaces on which the routing protocol is enabled.

Command Modes

OSPFv3 router configuration mode (config-router)

Command History

Release	Modification
15.1(3)S	This command was introduced.
Cisco IOS XE Release 3.4S	This command was integrated into Cisco IOS XE Release 3.4S.
15.2(1)T	This command was integrated into Cisco IOS Release 15.2(1)T.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

If you suppress the sending of routing updates on an interface, the particular address prefix will continue to be advertised to other interfaces, and updates from other routers on that interface continue to be received and processed.

The **default** keyword sets all interfaces as passive by default. You can then configure individual interfaces where adjacencies are desired using the **no passive-interface** command. The **default** keyword is useful in Internet service provider (ISP) and large enterprise networks where many of the distribution routers have more than 200 interfaces.

Examples

The following example sets all interfaces as passive, then activates Ethernet interface 0/0:

```
Router(config-router)# passive-interface default
Router(config-router)# no passive-interface ethernet0/0
```

Related Commands

Command	Description
default (OSPFv3)	Returns an OSPFv3 parameter to its default value.
router ospfv3	Enables OSPFv3 router configuration mode for the IPv4 or IPv6 address family.

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.

permit *protocol* {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] [**reflect** *name* [**timeout** *value*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

no permit *protocol* {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] [**reflect** *name* [**timeout** *value*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

Internet Control Message Protocol

permit icmp {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] [*icmp-type* [*icmp-code*]| *icmp-message*] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

Transmission Control Protocol

permit tcp {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] **[ack]** [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] **[established] [fin]** [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] [**neq** {*port*| *protocol*}] **[psh]** [**range** {*port*| *protocol*}] [**reflect** *name* [**timeout** *value*]] **[routing]** [**routing-type** *routing-number*] **[rst]** [**sequence** *value*] **[syn]** [**time-range** *name*] **[urg]**

User Datagram Protocol

permit udp {*source-ipv6-prefix/prefix-length*| **any**| **host** *source-ipv6-address*| **auth**} [*operator* [*port-number*]] {*destination-ipv6-prefix/prefix-length*| **any**| **host** *destination-ipv6-address*| **auth**} [*operator* [*port-number*]] [**dest-option-type** [*doh-number*| *doh-type*]] [**dscp** *value*] [**flow-label** *value*] **[fragments] [hbh] [log] [log-input] [mobility]** [**mobility-type** [*mh-number*| *mh-type*]] [**neq** {*port*| *protocol*}] [**range** {*port*| *protocol*}] [**reflect** *name* [**timeout** *value*]] **[routing]** [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*]

Syntax Description

protocol	Name or number of an Internet protocol. It can be one of the keywords **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **sctp**, **tcp**, **udp**, or **hbh**, 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 permit conditions. 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 *source-ipv6-address*	The source IPv6 host address about which to set permit conditions. 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.
auth	Allows matching traffic against the presence of the authentication header in combination with any protocol.
operator [*port-number*]	(Optional) Specifies an 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). If the operator is positioned after the *source-ipv6-prefix/prefix-length* argument, it must match the source port. If the operator is positioned after the *destination-ipv6-prefix/prefix-length* argument, it must match the destination port. The **range** operator requires two port numbers. All other operators require one port number. The optional *port-number* argument is a decimal number or the name of a TCP or UDP port. A port number is a number from 0 to 65535. 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	The destination IPv6 network or class of networks about which to set permit conditions. 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*	The destination IPv6 host address about which to set permit conditions. 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.
dest-option-type	(Optional) Matches IPv6 packets against the destination extension header within each IPv6 packet header.
doh-number	(Optional) Integer in the range from 0 to 255 representing an IPv6 destination option extension header.
doh-type	(Optional) Destination option header types. The possible destination option header type and its corresponding *doh-number* value are home-address—201.
dscp *value*	(Optional) Matches a differentiated services codepoint value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63.
flow-label *value*	(Optional) Matches a flow label value against the flow label value in the Flow Label field of each IPv6 packet header. The acceptable range is from 0 to 1048575.
fragments	(Optional) Matches non-initial fragmented packets where the fragment extension header contains a non-zero fragment offset. The **fragments** keyword is an option only if the *operator* [*port-number*] arguments are not specified. When this keyword is used, it also matches when the first fragment does not have Layer 4 information.
hbh	(Optional) Matches IPv6 packets against the hop-by-hop extension header within each IPv6 packet header.

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.) 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.
mobility	(mobility) Matches IPv6 packets against the mobility extension header within each IPv6 packet header.
mobility-type	(Optional) Matches IPv6 packets against the mobility-type extension header within each IPv6 packet header. Either the *mh-number* or *mh-type* argument must be used with this keyword.
mh-number	(Optional) Integer in the range from 0 to 255 representing an IPv6 mobility header type.
mh-type	(Optional) Mobility header types. Possible mobility header types and their corresponding *mh-number* value are as follows: • 0—bind-refresh • 1—hoti • 2—coti • 3—hot • 4—cot • 5—bind-update • 6—bind-acknowledgment • 7—bind-error

reflect *name*	(Optional) Specifies a reflexive IPvì6 access list. Reflexive IPv6 access lists are created dynamically when an IPv6 packets matches a permit statement that contains the **reflect** keyword. The reflexive IPv6 access list mirrors the permit statement and times out automatically when no IPv6 packets match the permit statement. Reflexive IPv6 access lists can be applied to the TCP, UDP, SCTP, and ICMP for IPv6 packets.
timeout *value*	(Optional) Interval of idle time (in seconds) after which a reflexive IPv6 access list times out. The acceptable range is from 1 to 4294967295. The default is 180 seconds.
routing	(Optional) Matches source-routed packets against the routing extension header within each IPv6 packet header.
routing-type	(Optional) Matches IPv6 packets against the routing-type extension header within each IPv6 packet header. The *routing-number* argument must be used with this keyword.
routing-number	Integer in the range from 0 to 255 representing an IPv6 routing header type. Possible routing header types and their corresponding *routing-number* value are as follows: • 0—Standard IPv6 routing header • 2—Mobile IPv6 routing header
sequence *value*	(Optional) Specifies the sequence number for the access list statement. The acceptable range is from 1 to 4294967295.
time-range *name*	(Optional) Specifies the time range that applies to the permit statement. The name of the time range and its restrictions are specified by the **time-range** and **absolute** or **periodic** commands, respectively.

icmp-type	(Optional) Specifies an ICMP message type for filtering ICMP packets. ICMP packets can be filtered by ICMP message type. The ICMP message type can be a number from 0 to 255, some of which include the following predefined strings and their corresponding numeric values: • 144—dhaad-request • 145—dhaad-reply • 146—mpd-solicitation • 147—mpd-advertisement
icmp-code	(Optional) Specifies an 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. The code is a number from 0 to 255.
icmp-message	(Optional) Specifies an ICMP message name for filtering ICMP packets. ICMP packets can be filtered by an ICMP message name or ICMP message type and code. The possible names are listed in the “Usage Guidelines” section.
ack	(Optional) For the TCP protocol only: acknowledgment (ACK) bit set.
established	(Optional) For the TCP protocol only: Indicates an established connection. A match occurs if the TCP datagram has the ACK or RST bits set. The nonmatching case is that of the initial TCP datagram to form a connection.
fin	(Optional) For the TCP protocol only: Fin bit set; no more data from sender.
neq {*port* | *protocol*}	(Optional) Matches only packets that are not on a given port number.
psh	(Optional) For the TCP protocol only: Push function bit set.
{**range** *port* | *protocol*}	(Optional) Matches only packets in the range of port numbers.
rst	(Optional) For the TCP protocol only: Reset bit set.
syn	(Optional) For the TCP protocol only: Synchronize bit set.

urg	(Optional) For the TCP protocol only: Urgent pointer bit set.

Command Default

No IPv6 access list is defined.

Command Modes

IPv6 access list configuration (config-ipv6-acl)#

Command History

Release	Modification
12.0(23)S	This command was introduced.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.4(2)T	The *icmp-type* argument was enhanced. The **dest-option-type**, **mobility**, **mobility-type**, and **routing-type** keywords were added. The *doh-number*, *doh-type*, *mh-number*, *mh-type*, and *routing-number* arguments were added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 series routers.
12.4(20)T	The **auth** keyword was added.
12.2(33)SRE	This command was integrated into Cisco IOS Release 12.2(33)SRE.
15.2(3)T	This command was modified. Support was added for the **hbh** keyword.
15.1(1)SY	This command was integrated into Cisco IOS Release 15.1(1)SY.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **permit** (IPv6) command is similar to the **permit** (IP) 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 or to define the access list as a reflexive 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 incremented by 10.

You can add **permit**, **deny**, **remark**, or **evaluate** 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.

In Cisco IOS Release 12.2(2)T or later releases, 12.0(21)ST, and 12.0(22)S, IPv6 access control lists (ACLs) are defined and their deny and permit conditions are set by using the **ipv6 access-list** command with the **deny** and **permit** keywords in global configuration mode. In Cisco IOS Release 12.0(23)S or later releases, IPv6 ACLs 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. Refer to the **ipv6 access-list** command for more information on defining IPv6 ACLs.

Note In Cisco IOS Release 12.0(23)S or later releases, 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. 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.

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.

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

The following is a list of ICMP message names:

- beyond-scope
- destination-unreachable
- echo-reply
- echo-request
- header
- hop-limit
- mld-query
- mld-reduction
- mld-report
- nd-na
- nd-ns

- next-header
- no-admin
- no-route
- packet-too-big
- parameter-option
- parameter-problem
- port-unreachable
- reassembly-timeout
- renum-command
- renum-result
- renum-seq-number
- router-advertisement
- router-renumbering
- router-solicitation
- time-exceeded
- unreachable

Defining Reflexive Access Lists

To define an IPv6 reflexive list, a form of session filtering, use the **reflect** keyword in the **permit** (IPv6) command. The **reflect** keyword creates an IPv6 reflexive access list and triggers the creation of entries in the reflexive access list. The **reflect** keyword must be an entry (condition statement) in an IPv6 access list.

Note For IPv6 reflexive access lists to work, you must nest the reflexive access list using the **evaluate** command.

If you are configuring IPv6 reflexive access lists for an external interface, the IPv6 access list should be one that is applied to outbound traffic.

If you are configuring an IPv6 reflexive access list for an internal interface, the IPv6 access list should be one that is applied to inbound traffic.

IPv6 sessions that originate from within your network are initiated with a packet exiting your network. When such a packet is evaluated against the statements in the IPv6 access list, the packet is also evaluated against the IPv6 reflexive permit entry.

As with all IPv6 access list entries, the order of entries is important, because they are evaluated in sequential order. When an IPv6 packet reaches the interface, it will be evaluated sequentially by each entry in the access list until a match occurs.

If the packet matches an entry prior to the reflexive permit entry, the packet will not be evaluated by the reflexive permit entry, and no temporary entry will be created for the reflexive access list (session filtering will not be triggered).

The packet will be evaluated by the reflexive permit entry if no other match occurs first. Then, if the packet matches the protocol specified in the reflexive permit entry, the packet is forwarded and a corresponding temporary entry is created in the reflexive access list (unless the corresponding entry already exists, indicating that the packet belongs to a session in progress). The temporary entry specifies criteria that permit traffic into your network only for the same session.

Characteristics of Reflexive Access List Entries

The **permit** (IPv6) command with the **reflect** keyword enables the creation of temporary entries in the same IPv6 reflexive access list that was defined by the **permit** (IPv6) command. The temporary entries are created when an IPv6 packet exiting your network matches the protocol specified in the **permit** (IPv6) command. (The packet “triggers” the creation of a temporary entry.) These entries have the following characteristics:

- The entry is a permit entry.
- The entry specifies the same IP upper-layer protocol as the original triggering packet.
- The entry specifies the same source and destination addresses as the original triggering packet, except that the addresses are swapped.
- If the original triggering packet is TCP or UDP, the entry specifies the same source and destination port numbers as the original packet, except that the port numbers are swapped.
- If the original triggering packet is a protocol other than TCP or UDP, port numbers do not apply, and other criteria are specified. For example, for ICMP, type numbers are used: The temporary entry specifies the same type number as the original packet (with only one exception: if the original ICMP packet is type 8, the returning ICMP packet must be type 0 to be matched).
- The entry inherits all the values of the original triggering packet, with exceptions only as noted in the previous four bullets.
- IPv6 traffic entering your internal network will be evaluated against the entry, until the entry expires. If an IPv6 packet matches the entry, the packet will be forwarded into your network.
- The entry will expire (be removed) after the last packet of the session is matched.
- If no packets belonging to the session are detected for a configured length of time (the timeout period), the entry will expire.

Examples

The following example configures two IPv6 access lists named OUTBOUND and INBOUND and applies both access lists to outbound and inbound traffic on Ethernet interface 0. The first and second permit entries in the OUTBOUND list permit all TCP and UDP packets from network 2001:ODB8:0300:0201::/64 to exit out of Ethernet interface 0. The entries also configure the temporary IPv6 reflexive access list named REFLECTOUT to filter returning (incoming) TCP and UDP packets on Ethernet interface 0. The first deny entry in the OUTBOUND list keeps all packets from the network FEC0:0:0:0201::/64 (packets that have the site-local prefix FEC0:0:0:0201 as the first 64 bits of their source IPv6 address) from exiting out of Ethernet interface 0. The third permit entry in the OUTBOUND list permits all ICMP packets to exit out of Ethernet interface 0.

The permit entry in the INBOUND list permits all ICMP packets to enter Ethernet interface 0. The **evaluate** command in the list applies the temporary IPv6 reflexive access list named REFLECTOUT to inbound TCP and UDP packets on Ethernet interface 0. When outgoing TCP or UDP packets are permitted on Ethernet interface 0 by the OUTBOUND list, the INBOUND list uses the REFLECTOUT list to match (evaluate) the returning (incoming) TCP and UDP packets. Refer to the **evaluate** command for more information on nesting IPv6 reflexive access lists within IPv6 ACLs.

```
ipv6 access-list OUTBOUND
 permit tcp 2001:0DB8:0300:0201::/64 any reflect REFLECTOUT
 permit udp 2001:0DB8:0300:0201::/64 any reflect REFLECTOUT
 deny FEC0:0:0:0201::/64 any
 permit icmp any any
ipv6 access-list INBOUND
 permit icmp any any
 evaluate REFLECTOUT
interface ethernet 0
 ipv6 traffic-filter OUTBOUND out
 ipv6 traffic-filter INBOUND in
```

Note Given that a **permit any any** statement is not included as the last entry in the OUTBOUND or INBOUND access list, only TCP, UDP, and ICMP packets will be permitted out of and in to Ethernet interface 0 (the implicit deny all condition at the end of the access list denies all other packet types on the interface).

The following example shows how to allow the matching of any UDP traffic. The authentication header may be present.

```
permit udp any any sequence 10
```

The following example shows how to allow the matching of only TCP traffic if the authentication header is also present.

```
permit tcp any any auth sequence 20
```

The following example shows how to allow the matching of any IPv6 traffic where the authentication header is present.

```
permit ahp any any sequence 30
```

Related Commands

Command	Description
deny (IPv6)	Sets deny conditions for an IPv6 access list.
evaluate (IPv6)	Nests an IPv6 reflexive access list within an IPv6 access list.
ipv6 access-list	Defines an IPv6 access list and enters IPv6 access list configuration mode.
ipv6 traffic-filter	Filters incoming or outgoing IPv6 traffic on an interface.
show ipv6 access-list	Displays the contents of all current IPv6 access lists.

prefix-glean

To enable the device to glean prefixes from IPv6 router advertisements (RAs) or Dynamic Host Configuration Protocol (DHCP), use the **prefix-glean** command in IPv6 snooping configuration mode. To learn only prefixes gleaned in one of these protocols and exclude the other, use the **no** form of this command.

prefix-glean [**only**]

no prefix-glean [**only**]

Syntax Description

only	(Optional) Only prefixes are gleaned.

Command Default

Prefixes are not learned through RA or DHCP.

Command Modes

IPv6 snooping configuration mode (config-ipv6-snooping)

Command History

Release	Modification
15.0(2)SE	This command was introduced.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **prefix-glean** command enables the device to learn prefixes in RA and DHCP traffic.

Examples

The following example shows how to enable the device to learn prefixes:

```
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# prefix-glean
```

Related Commands

Command	Description
ipv6 snooping attach-policy	Applies an IPv6 snooping policy to a target.
ipv6 snooping policy	Configures an IPv6 snooping policy and enters IPv6 snooping configuration mode.

protocol (IPv6)

To specify that addresses should be gleaned with Dynamic Host Configuration Protocol (DHCP) or Neighbor Discovery Protocol (NDP) or to associate the protocol with an IPv6 prefix list, use the **protocol** command. To disable address gleaning with DHCP or NDP, use the **no** form of the command.

protocol {**dhcp** | **ndp**} [**prefix-list** *prefix-list-name*]

no protocol {**dhcp** | **ndp**}

Syntax Description

dhcp	Specifies that addresses should be gleaned in Dynamic Host Configuration Protocol (DHCP) packets.
ndp	Specifies that addresses should be gleaned in Neighbor Discovery Protocol (NDP) packets.
prefix-list *prefix-list-name*	(Optional) Specifies that a prefix list of protected prefixes be used.

Command Default

Snooping and recovery are attempted using both DHCP and NDP. No prefix list is used, all address ranges are accepted.

Command Modes

IPv6 snooping configuration mode (config-ipv6-snooping)

Command History

Release	Modification
15.2(4)S	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

If an address does not match the prefix list associated with DHCP or NDP, then control packets will be dropped and recovery of the binding table entry will not be attempted with that protocol.

- If there is no prefix list specified, all protocols are supported by default. There is no check and all addresses are accepted.
- Using the **no protocol** {**dhcp** | **ndp**} command indicates that a protocol will not to be used for snooping or gleaning.
- However, if the **no protocol dhcp** command is used, DHCP can still be used for binding table recovery.
- The NDP prefix list should be a superset of the DHCP prefix list, as addresses obtained by DHCP must be confirmed by NDP later.

- When a prefix list is given and a protocol packet indicates an address that does not match the prefix list for that protocol, the packet is dropped (unless the security level is “glean”).
- Data glean can recover with DHCP and NDP, though destination guard will only recovery through DHCP.

Note Before you configure the **protocol** command, it is essential that you provide a value for the **ge** *ge-value* option when configuring a prefix list using the **ipv6 prefix-list** command.

Examples

The following example shows a valid configuration for an IPv6 prefix list (“abc”) and shows that DHCP will be used to recover addresses that match the prefix list abc:

```
Device(config)# ipv6 prefix-list abc seq 5 permit 2001:DB8::/64 ge 128
!
Device(config-ipv6-snooping)# protocol dhcp prefix-list abc
```

Related Commands

Command	Description
ipv6 prefix-list	Creates an entry in an IPv6 prefix list.
ipv6 snooping policy	Enters IPv6 snooping configuration mode.

redistribute (IPv6)

To redistribute IPv6 routes from one routing domain into another routing domain, use the **redistribute** command in address family configuration or router configuration mode. To disable redistribution, use the **no** form of this command.

redistribute source-protocol [*process-id*] [**include-connected** {**level-1**| **level-1-2**| **level-2**}] [*as-number*] [**metric** {*metric-value*| **transparent**}] [**metric-type** *type-value*] [**match** {**external** [**1**| **2**]| **internal**| **nssa-external** [**1**| **2**]}] [**tag** *tag-value*] [**route-map** *map-tag*]

no redistribute source-protocol [*process-id*] **[include-connected]** {**level-1**| **level-1-2**| **level-2**} [*as-number*] [**metric** {*metric-value*| **transparent**}] [**metric-type** *type-value*] [**match** {**external** [**1**| **2**]| **internal**| **nssa-external** [**1**| **2**]}] [**tag** *tag-value*] [**route-map** *map-tag*]

Syntax Description

source-protocol	Source protocol from which routes are being redistributed. It can be one of the following keywords: **bgp**, connected, **eigrp**, **isis**, **ospf**, **rip**, or **static**.
process-id	(Optional) For the **bgp** or **eigrp**keyword, the process ID is a Border Gateway Protocol (BGP) autonomous system number, which is a 16-bit decimal number. For the **isis**keyword, the process ID is an optional value that defines a meaningful name for a routing process. You can specify only one IS-IS process per router. Creating a name for a routing process means that you use names when configuring routing. For the **ospf** keyword, the process ID is the number assigned administratively when the Open Shortest Path First (OSPF) for IPv6 routing process is enabled. For the **rip**keyword, the process ID is an optional value that defines a meaningful name for an IPv6 Routing Information Protocol (RIP) routing process.
include-connected	(Optional) Allows the target protocol to redistribute routes learned by the source protocol and connected prefixes on those interfaces over which the source protocol is running.
level-1	Specifies that, for Intermediate System-to-Intermediate System (IS-IS), Level 1 routes are redistributed into other IP routing protocols independently.
level-1-2	Specifies that, for IS-IS, both Level 1 and Level 2 routes are redistributed into other IP routing protocols.

level-2	Specifies that, for IS-IS, Level 2 routes are redistributed into other IP routing protocols independently.
as-number	(Optional) Autonomous system number for the redistributed route.
metric *metric-value*	(Optional) When redistributing from one OSPF process to another OSPF process on the same router, the metric will be carried through from one process to the other if no metric value is specified. When redistributing other processes to an OSPF process, the default metric is 20 when no metric value is specified.
metric transparent	(Optional) Causes RIP to use the routing table metric for redistributed routes as the RIP metric.
metric-type *type-value*	(Optional) For OSPF, specifies the external link type associated with the default route advertised into the OSPF routing domain. It can be one of two values: • **1** --Type 1 external route • **2** --Type 2 external route If no value is specified for the **metric-type** keyword, the Cisco IOS software adopts a Type 2 external route. For IS-IS, the link type can be one of two values: • **internal** --IS-IS metric that is < 63. • **external** --IS-IS metric that is > 64 < 128. The default is **internal**.
match {**external [1 | 2]| internal | nssa-external [1 | 2]**	(Optional) For OSPF, routes are redistributed into other routing domains using the match keyword. It is used with one of the following: • **external [1 | 2]** --Routes that are external to the autonomous system, but are imported into OSPF as Type 1 or Type 2 external routes. • **internal** --Routes that are internal to a specific autonomous system. • **nssa-external** [**1**| **2**]-- Routes that are external to the autonomous system but are imported into OSPF, in a not so stubby area (NSSA), for IPv6 as Type 1 or Type 2 external routes.

tag *tag-value*	(Optional) Specifies the 32-bit decimal value attached to each external route. This is not used by OSPF itself. It may be used to communicate information between Autonomous System Boundary Routers (ASBRs). If none is specified, then the remote autonomous system number is used for routes from BGP and Exterior Gateway Protocol (EGP); for other protocols, zero (0) is used.
route-map	(Optional) Specifies the route map that should be checked to filter the importation of routes from this source routing protocol to the current routing protocol. If the route-map keyword is not specified, all routes are redistributed. If this keyword is specified, but no route map tags are listed, no routes will be imported.
map-tag	(Optional) Identifier of a configured route map.

Command Default

Route redistribution is disabled.

Command Modes

Address family configuration Router configuration

Command History

Release	Modification
12.2(15)T	This command was introduced.
12.4(6)T	Support for Enhanced Internal Gateway Routing Protocol (EIGRP) IPv6 was added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Series Routers.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

Changing or disabling any keyword will not affect the state of other keywords.

A router receiving an IPv6 IS-IS route with an internal metric will consider the cost of the route from itself to the redistributing router plus the advertised cost to reach the destination. An external metric considers only the advertised metric to reach the destination.

IS-IS will ignore any configured redistribution of routes configured with the include-connected keyword. IS-IS will advertise a prefix on an interface if either IS-IS is running over the interface or the interface is configured as passive.

Routes learned from IPv6 routing protocols can be redistributed into IPv6 IS-IS at Level 1 into an attached area or at Level 2. The **level-1-2** keyword allows both Level 1 and Level 2 routes in a single command.

For IPv6 RIP, use the **redistribute**command to advertise static routes as if they were directly connected routes.

Caution Advertising static routes as directly connected routes can cause routing loops if improperly configured.

Redistributed IPv6 RIP routing information should always be filtered by the **distribute-list prefix-list**router configuration command. Use of the **distribute-list prefix-list**command ensures that only those routes intended by the administrator are passed along to the receiving routing protocol.

Note The **metric** value specified in the **redistribute** command for IPv6 RIP supersedes the **metric** value specified using the **default-metric** command.

Note In IPv4, if you redistribute a protocol, by default you also redistribute the subnet on the interfaces over which the protocol is running. In IPv6 this is not the default behavior. To redistribute the subnet on the interfaces over which the protocol is running in IPv6, use the include-connected keyword. In IPv6 this functionality is not supported when the source protocol is BGP.

When the no redistribute command is configured, the parameter settings are ignored when the client protocol is IS-IS or EIGRP.

IS-IS redistribution will be removed completely when IS-IS level 1 and level 2 are removed by the user. IS-IS level settings can be configured using the redistribute command only.

The default redistribute type will be restored to OSPF when all route type values are removed by the user.

Examples

The following example configures IPv6 IS-IS to redistribute IPv6 BGP routes. The metric is specified as 5, and the metric type will be set to external, indicating that it has lower priority than internal metrics.

```
Router(config)# router isis
Router(config-router)# address-family ipv6
Router(config-router-af)# redistribute bgp 64500 metric 5 metric-type external
```

The following example redistributes IPv6 BGP routes into the IPv6 RIP routing process named cisco:

```
Router(config)# ipv6 router rip cisco
Router(config-router)# redistribute bgp 42
```

The following example redistributes IS-IS for IPv6 routes into the OSPF for IPv6 routing process 1:

```
Router(config)# ipv6 router ospf 1
Router(config-router)# redistribute isis 1 metric 32 metric-type 1 tag 85
```

In the following example, ospf 1 redistributes the prefixes 2001:1:1::/64 and 2001:99:1::/64 and any prefixes learned through rip 1:

```
interface ethernet0/0
 ipv6 address 2001:1:1::90/64
 ipv6 rip 1 enable
interface ethernet1/1
 ipv6 address 2001:99:1::90/64
 ipv6 rip 1 enable
interface ethernet2/0
 ipv6 address 2001:1:2::90/64
 ipv6 ospf 1 area 1
ipv6 router ospf 1
  redistribute rip 1 include-connected
```

The following configuration example and output show the no redistribute command parameters when the last route type value is removed:

```
Router(config-router)# redistribute rip process1 metric 7
Router(config-router)# do show run | include redistribute
 redistribute rip process1 metric 7
Router(config-router)# no redistribute rip process1 metric 7
Router(config-router)# do show run | include redistribute
 redistribute rip process1
Router(config-router)#
```

Related Commands

Command	Description
default-metric	Specifies a default metric for redistributed routes.
distribute-list prefix-list (IPv6 EIGRP)	Applies a prefix list to EIGRP for IPv6 routing updates that are received or sent on an interface.
distribute-list prefix-list (IPv6 RIP)	Applies a prefix list to IPv6 RIP routing updates that are received or sent on an interface.
redistribute isis (IPv6)	Redistributes IPv6 routes from one routing domain into another routing domain using IS-IS as both the target and source protocol.

router-preference maximum

To verify the advertised default router preference parameter value, use the **router-preference maximum** command in RA guard policy configuration mode.

router-preference maximum {**high**| **low**| **medium**}

Syntax Description

high	Default router preference parameter value is higher than the specified limit.
medium	Default router preference parameter value is equal to the specified limit.
low	Default router preference parameter value is lower than the specified limit.

Command Default

The router preference maximum value is not configured.

Command Modes

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **router-preference maximum** command enables verification that the advertised default router preference parameter value is lower than or equal to a specified limit. You can use this command to give a lower priority to default routers advertised on trunk ports, and to give precedence to default routers advertised on access ports.

The **router-preference maximum** command limit are high, medium, or low. If, for example, this value is set to **medium** and the advertised default router preference is set to **high** in the received packet, then the packet is dropped. If the command option is set to **medium** or **low** in the received packet, then the packet is not dropped.

Examples

The following example shows how the command defines a router advertisement (RA) guard policy name as raguard1, places the router in RA guard policy configuration mode, and configures router-preference maximum verification to be high:

```
Router(config)# ipv6 nd raguard policy raguard1
Router(config-ra-guard)# router-preference maximum high
```

Related Commands

Command	Description
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.

ipv6-r1

sec-level minimum

To specify the minimum security level parameter value when Cryptographically Generated Address (CGA) options are used, use the **sec-level minimum** command in Neighbor Discovery (ND) inspection policy configuration mode. To disable this function, use the **no** form of this command.

sec-level minimum *value*

no sec-level minimum *value*

Syntax Description

value	Minimum security level, which is a value from 1 to 7. The default security level is 1. The most secure level is 3.

Command Default

The default security level is 1.

Command Modes

ND inspection policy configuration (config-nd-inspection)

RA guard policy configuration (config-ra-guard)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **sec-level minimum** command specifies the minimum security level parameter value when CGA options are used. Use the **sec-level minimum** command after enabling ND inspection policy configuration mode using the **ipv6 nd inspection policy** command.

Examples

The following example defines an ND policy name as policy1, places the router in ND inspection policy configuration mode, and specifies 2 as the minimum CGA security level:

```
Router(config)# ipv6 nd inspection policy policy1
Router(config-nd-inspection)# sec-level minimum 2
```

Related Commands

Command	Description
ipv6 nd inspection policy	Defines the ND inspection policy name and enters ND inspection policy configuration mode.
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.

server name (IPv6 TACACS+)

To specify an IPv6 TACACS+ server, use the **server name**command in TACACS+ group server configuration mode. To remove the IPv6 TACACS+ server from configuration, use the **no** form of this command.

server name *server-name*

no server name *server-name*

Syntax Description

server-name	The IPv6 TACACS+ server to be used.

Command Default

No server name is specified.

Command Modes

TACACS+ group server configuration (config-sg-tacacs+)

Command History

Release	Modification
Cisco IOS XE Release 3.2S	This command was introduced.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

You must configure the **aaa group server tacacs** command before configuring this command.

Enter the **server name** command to specify an IPv6 TACACS+ server.

Examples

The following example shows how to specify an IPv6 TACACS+ server named server1:

```
Router(config)# aaa group server tacacs+
Router(config-sg-tacacs+)# server name server1
```

Related Commands

Command	Description
aaa group server tacacs	Configures the TACACS+ server for IPv6 or IPv4 and enters TACACS+ server configuration mode.

show ipv6 access-list

To display the contents of all current IPv6 access lists, use the **show ipv6 access-list**command in user EXEC or privileged EXEC mode.

show ipv6 access-list [*access-list-name*]

Syntax Description

access-list-name	(Optional) Name of access list.

Command Default

All IPv6 access lists are displayed.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.0(23)S	The priority field was changed to sequence and Layer 4 protocol information (extended IPv6 access list functionality) was added to the display output.
12.2(13)T	This command was integrated into Cisco IOS Release 12.2(13)T.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.2(50)SY	This command was modified. Information about IPv4 and IPv6 hardware statistics is displayed.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **show ipv6 access-list** command provides output similar to the **show ip access-list** command, except that it is IPv6-specific.

Examples

The following output from the **show ipv6 access-list**command shows IPv6 access lists named inbound, tcptraffic, and outbound:

```
Router# show ipv6 access-list
IPv6 access list inbound
    permit tcp any any eq bgp reflect tcptraffic (8 matches) sequence 10
    permit tcp any any eq telnet reflect tcptraffic (15 matches) sequence 20
    permit udp any any reflect udptraffic sequence 30
IPv6 access list tcptraffic (reflexive) (per-user)
    permit tcp host 2001:0DB8:1::1 eq bgp host 2001:0DB8:1::2 eq 11000 timeout 300 (time
       left 243) sequence 1
    permit tcp host 2001:0DB8:1::1 eq telnet host 2001:0DB8:1::2 eq 11001 timeout 300
    (time left 296) sequence 2
IPv6 access list outbound
    evaluate udptraffic
    evaluate tcptraffic
```

The following sample output shows IPv6 access list information for use with IPSec:

```
Router#  show ipv6 access-list
IPv6 access list Tunnel0-head-0-ACL (crypto)
     permit ipv6 any any (34 matches) sequence 1
IPv6 access list Ethernet2/0-ipsecv6-ACL (crypto)
     permit 89 FE80::/10 any (85 matches) sequence 1
```

The table below describes the significant fields shown in the display.

Table 1: show ipv6 access-list Field Descriptions

Field	Description
ipv6 access list inbound	Name of the IPv6 access list, for example, inbound.
permit	Permits any packet that matches the specified protocol type.
tcp	Transmission Control Protocol. The higher-level (Layer 4) protocol type that the packet must match.
any	Equal to ::/0.
eq	An equal operand that compares the source or destination ports of TCP or UDP packets.
bgp	Border Gateway Protocol. The lower-level (Layer 3) protocol type that the packet must be equal to.
reflect	Indicates a reflexive IPv6 access list.

Field	Description
tcptraffic (8 matches)	The name of the reflexive IPv6 access list and the number of matches for the access list. The **clear ipv6 access-list** privileged EXEC command resets the IPv6 access list match counters.
sequence 10	Sequence in which an incoming packet is compared to lines in an access list. Lines in an access list are ordered from first priority (lowest number, for example, 10) to last priority (highest number, for example, 80).
host 2001:0DB8:1::1	The source IPv6 host address that the source address of the packet must match.
host 2001:0DB8:1::2	The destination IPv6 host address that the destination address of the packet must match.
11000	The ephemeral source port number for the outgoing connection.
timeout 300	The total interval of idle time (in seconds) after which the temporary IPv6 reflexive access list named tcptraffic will time out for the indicated session.
(time left 243)	The amount of idle time (in seconds) remaining before the temporary IPv6 reflexive access list named tcptraffic is deleted for the indicated session. Additional received traffic that matches the indicated session resets this value to 300 seconds.
evaluate udptraffic	Indicates the IPv6 reflexive access list named udptraffic is nested in the IPv6 access list named outbound.

Related Commands

Command	Description
clear ipv6 access-list	Resets the IPv6 access list match counters.
hardware statistics	Enables the collection of hardware statistics.
show ip access-list	Displays the contents of all current IP access lists.
show ip prefix-list	Displays information about a prefix list or prefix list entries.

Command	Description
show ipv6 prefix-list	Displays information about an IPv6 prefix list or IPv6 prefix list entries.

show ipv6 dhcp conflict

To display address conflicts found by a Dynamic Host Configuration Protocol for IPv6 (DHCPv6) server when addresses are offered to the client, use the **show ipv6 dhcp conflict** command in privileged EXEC mode.

show ipv6 dhcp conflict [*ipv6-address*] [**vrf** *vrf-name*]

Syntax Description

ipv6-address	(Optional) The address of a DHCP for IPv6 client.
vrf *vrf-name*	(Optional) Specifies a virtual routing and forwarding (VRF) configuration.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.4(24)T	This command was introduced.
Cisco IOS XE Release 2.5	This command was integrated into Cisco IOS XE Release 2.5.
15.1(2)S	This command was modified. The **vrf** *vrf-name* keyword and argument were added.
Cisco IOS XE Release 3.3S	This command was modified. The **vrf** *vrf-name* keyword and argument were added.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

When you configure the DHCPv6 server to detect conflicts, it uses ping. The client uses neighbor discovery to detect clients and reports to the server through a DECLINE message. If an address conflict is detected, the address is removed from the pool, and the address is not assigned until the administrator removes the address from the conflict list.

Examples

The following is a sample output from the **show ipv6 dhcp conflict** command. This command shows the pool and prefix values for DHCP conflicts.:

```
Router# show ipv6 dhcp conflict
Pool 350, prefix 2001:0DB8:1005::/48
     2001:0DB8:1005::10
```

Related Commands

Command	Description
clear ipv6 dhcp conflict	Clears an address conflict from the DHCPv6 server database.

show ipv6 interface

To display the usability status of interfaces configured for IPv6, use the **show ipv6 interface**command in user EXEC or privileged EXEC mode.

show ipv6 interface [brief] [*type number*] **[prefix]**

Syntax Description

brief	(Optional) Displays a brief summary of IPv6 status and configuration for each interface.
type	(Optional) The interface type about which to display information.
number	(Optional) The interface number about which to display information.
prefix	(Optional) Prefix generated from a local IPv6 prefix pool.

Command Default

All IPv6 interfaces are displayed.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.2(4)T	The OK, TENTATIVE, DUPLICATE, ICMP redirects, and ND DAD fields were added to the command output.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(25)S	Command output was updated to display information on the current Unicast RPF configuration.
12.4(2)T	Command output was updated to show the state of the default router preference (DRP) preference value as advertised by a device through an interface.

Release	Modification
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.4(4)T	Command output was updated to show Hot Standby Router Protocol (HSRP) for IPv6 information.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 series devices.
12.4(24)T	Command output was updated to show the Dynamic Host Configuration Protocol (DHCP) originated addresses.
12.2(50)SY	This command was integrated into Cisco IOS Release 12.2(50)SY.
15.0(1)SY	This command was integrated into Cisco IOS Release 15.0(1)SY.
15.2(2)SNG	This command was implemented on the Cisco ASR 901 Series Aggregation Services devices.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **show ipv6 interface** command provides output similar to the show ip interface command, except that it is IPv6-specific.

Use the **show ipv6 interface** command to validate the IPv6 status of an interface and its configured addresses. The show ipv6 interface command also displays the parameters that IPv6 is using for operation on this interface and any configured features.

If the interface’s hardware is usable, the interface is marked up. If the interface can provide two-way communication for IPv6, the line protocol is marked up.

If you specify an optional interface type and number, the command displays information only about that specific interface. For a specific interface, you can enter the prefix keyword to see the IPv6 neighbor discovery (ND) prefixes that are configured on the interface.

Examples

Examples

The **show ipv6 interface**command displays information about the specified interface.

```
Device(config)# show ipv6 interface ethernet0/0
Ethernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::A8BB:CCFF:FE00:6700
  No Virtual link-local address(es):
  Global unicast address(es):
    2001::1, subnet is 2001::/64 [DUP]
    2001::A8BB:CCFF:FE00:6700, subnet is 2001::/64 [EUI]
    2001:100::1, subnet is 2001:100::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:1
    FF02::1:FF00:6700
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 30000)
  ND advertised reachable time is 0 (unspecified)
  ND advertised retransmit interval is 0 (unspecified)
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  ND advertised default router preference is Medium
  Hosts use stateless autoconfig for addresses.
```

The table below describes the significant fields shown in the display.

Table 2: show ipv6 interface Field Descriptions

Field	Description
Ethernet0/0 is up, line protocol is up	Indicates whether the interface hardware is 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, down (down is not shown in sample output)	Indicates whether the software processes that handle the line protocol consider the line usable (that is, whether keepalives are successful or IPv6 CP has been negotiated). 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.
Global unicast address(es):	Displays the global unicast addresses assigned to the interface.
Joined group address(es):	Indicates the multicast groups to which this interface belongs.

Field	Description
MTU	Maximum transmission unit of the interface.
ICMP error messages	Specifies the minimum interval (in milliseconds) between error messages sent on this interface.
ICMP redirects	The state of Internet Control Message Protocol (ICMP) IPv6 redirect messages on the interface (the sending of the messages is enabled or disabled).
ND DAD	The 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.
ND advertised reachable time	Displays the neighbor discovery reachable time (in milliseconds) advertised on this interface.
ND advertised retransmit interval	Displays the neighbor discovery retransmit interval (in milliseconds) advertised on this interface.
ND router advertisements	Specifies the interval (in seconds) for neighbor discovery router advertisements (RAs) sent on this interface and the amount of time before the advertisements expire. As of Cisco IOS Release 12.4(2)T, this field displays the default router preference (DRP) value sent by this device on this interface.
ND advertised default router preference is Medium	The DRP for the device on a specific interface.

Examples

The **show ipv6 interface** command displays information about attributes that may be associated with an IPv6 address assigned to the interface.

Attribute	Description
ANY	Anycast. The address is an anycast address, as specified when configured using the **ipv6 address** command.
CAL	Calendar. The address is timed and has valid and preferred lifetimes.

Attribute	Description
DEP	Deprecated. The timed address is deprecated.
DUP	Duplicate. The address is a duplicate, as determined by duplicate address detection (DAD). To re-attampt DAD, the user must use the **shutdown** or **no shutdown** command on the interface.
EUI	EUI-64 based. The address was generated using EUI-64.
OFF	Offlink. The address is offlink.
OOD	Overly optimistic DAD. DAD will not be performed for this address. This attribute applies to virtual addresses.
PRE	Preferred. The timed address is preferred.
TEN	Tentative. The address is in a tentative state per DAD.
UNA	Unactivated. The virtual address is not active and is in a standby state.
VIRT	Virtual. The address is virtual and is managed by HSRP, VRRP, or GLBP.

The following is sample output from the **show ipv6 interface**command when entered with the **brief** keyword:

```
Device# show ipv6 interface brief
Ethernet0 is up, line protocol is up
Ethernet0                  [up/up]
    unassigned
Ethernet1                  [up/up]
    2001:0DB8:1000:/29
Ethernet2                  [up/up]
    2001:0DB8:2000:/29
Ethernet3                  [up/up]
    2001:0DB8:3000:/29
Ethernet4                  [up/down]
    2001:0DB8:4000:/29
Ethernet5                  [administratively down/down]
    2001:123::210:7BFF:FEC2:ACD8
Interface          Status                 IPv6 Address
Ethernet0          up                     3FFE:C00:0:1:260:3EFF:FE11:6770
Ethernet1          up                     unassigned
Fddi0              up                     3FFE:C00:0:2:260:3EFF:FE11:6772
Serial0            administratively down unassigned
Serial1            administratively down unassigned
Serial2            administratively down unassigned
Serial3            administratively down unassigned
Tunnel0            up                     unnumbered (Ethernet0)
Tunnel1            up                     3FFE:700:20:1::12
```

Examples

This sample output shows the characteristics of an interface that has generated a prefix from a local IPv6 prefix pool:

```
Device# show ipv6 interface Ethernet 0/0 prefix

interface Ethernet0/0
 ipv6 address 2001:0DB8::1/64
 ipv6 address 2001:0DB8::2/64
 ipv6 nd prefix 2001:0DB8:2::/64
 ipv6 nd prefix 2001:0DB8:3::/64 2592000 604800 off-link
end
.
.
.
IPv6 Prefix Advertisements Ethernet0/0
Codes: A - Address, P - Prefix-Advertisement, O - Pool
       U - Per-user prefix, D - Default
       N - Not advertised, C - Calendar
     default [LA] Valid lifetime 2592000, preferred lifetime 604800
AD   2001:0DB8:1::/64 [LA] Valid lifetime 2592000, preferred lifetime 604800
APD  2001:0DB8:2::/64 [LA] Valid lifetime 2592000, preferred lifetime 604800
P    2001:0DB8:3::/64 [A] Valid lifetime 2592000, preferred lifetime 604800
```

The default prefix shows the parameters that are configured using the ipv6 nd prefix default command.

Examples

This sample output shows the state of the DRP preference value as advertised by this device through an interface:

```
Device# show ipv6 interface gigabitethernet 0/1
  GigabitEthernet0/1 is up, line protocol is up
    IPv6 is enabled, link-local address is FE80::130
    Description: Management network (dual stack)
    Global unicast address(es):
      FEC0:240:104:1000::130, subnet is FEC0:240:104:1000::/64
    Joined group address(es):
      FF02::1
      FF02::2
      FF02::1:FF00:130
    MTU is 1500 bytes
    ICMP error messages limited to one every 100 milliseconds
    ICMP redirects are enabled
    ND DAD is enabled, number of DAD attempts: 1
    ND reachable time is 30000 milliseconds
    ND advertised 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
    ND advertised default router preference is Low
    Hosts use stateless autoconfig for addresses.
```

Examples

When HSRP IPv6 is first configured on an interface, the interface IPv6 link-local address is marked unactive (UNA) because it is no longer advertised, and the HSRP IPv6 virtual link-local address is added to the virtual link-local address list with the UNA and tentative DAD (TEN) attributes set. The interface is also programmed to listen for the HSRP IPv6 multicast address.

This sample output shows the status of UNA and TEN attributes, when HSRP IPv6 is configured on an interface:

```
Device# show ipv6 interface ethernet 0/0
Ethernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80:2::2 [UNA]
  Virtual link-local address(es):
    FE80::205:73FF:FEA0:1 [UNA/TEN]
  Global unicast address(es):
    2001:2::2, subnet is 2001:2::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::66
    FF02::1:FF00:2
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ND DAD is enabled, number of DAD attempts: 1
```

After the HSRP group becomes active, the UNA and TEN attributes are cleared, and the overly optimistic DAD (OOD) attribute is set. The solicited node multicast address for the HSRP virtual IPv6 address is also added to the interface.

This sample output shows the status of UNA, TEN and OOD attributes, when HSRP group is activated:

```
Device# show ipv6 interface ethernet 0/0
Ethernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80:2::2 [UNA]
  Virtual link-local address(es):
    FE80::205:73FF:FEA0:1 [OPT]
  Global unicast address(es):
    2001:2::2, subnet is 2001:2::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::66
    FF02::1:FF00:2
    FF02::1:FFA0:1
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
```

The table below describes additional significant fields shown in the displays for the **show ipv6 interface** command with HSRP configured.

Table 3: show ipv6 interface Command with HSRP Configured Field Descriptions

Field	Description
IPv6 is enabled, link-local address is FE80:2::2 [UNA]	The interface IPv6 link-local address is marked UNA because it is no longer advertised.
FE80::205:73FF:FEA0:1 [UNA/TEN]	The virtual link-local address list with the UNA and TEN attributes set.
FF02::66	HSRP IPv6 multicast address.
FE80::205:73FF:FEA0:1 [OPT]	HSRP becomes active, and the HSRP virtual address marked OPT.
FF02::1:FFA0:1	HSRP solicited node multicast address.

Examples

When you enable Mobile IPv6 on an interface, you can configure a minimum interval between IPv6 router advertisement (RA) transmissions. The **show ipv6 interface** command output reports the minimum RA interval, when configured. If the minimum RA interval is not explicitly configured, then it is not displayed.

In the following example, the maximum RA interval is configured as 100 seconds, and the minimum RA interval is configured as 60 seconds on Ethernet interface 1/0:

```
Device(config-if)# ipv6 nd ra-interval 100 60
```

Subsequent use of the **show ipv6 interface** then displays the interval as follows:

```
Device(config)# show ipv6 interface ethernet 1/0
Ethernet1/0 is administratively down, line protocol is down
  IPv6 is enabled, link-local address is FE80::A8BB:CCFF:FE00:5A01 [TEN]
  No Virtual link-local address(es):
  No global unicast address is configured
  Joined group address(es):
    FF02::1
    FF02::2
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 60 to 100 seconds
  ND router advertisements live for 1800 seconds
  ND advertised default router preference is Medium
  Hosts use stateless autoconfig for addresses.
```

In the following example, the maximum RA interval is configured as 100 milliseconds (ms), and the minimum RA interval is configured as 60 ms on Ethernet interface 1/0:

```
Device(config)# show ipv6 interface ethernet 1/0
Ethernet1/0 is administratively down, line protocol is down
  IPv6 is enabled, link-local address is FE80::A8BB:CCFF:FE00:5A01 [TEN]
  No Virtual link-local address(es):
  No global unicast address is configured
  Joined group address(es):
    FF02::1
    FF02::2
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised reachable time is 0 milliseconds
  ND advertised retransmit interval is 0 milliseconds
  ND router advertisements are sent every 60 to 100 milliseconds
  ND router advertisements live for 1800 seconds
  ND advertised default router preference is Medium
  Hosts use stateless autoconfig for addresses.
```

The table below describes additional significant fields shown in the displays for the **show ipv6 interface** command with minimum RA interval information configured.

Table 4: show ipv6 interface Command with Minimum RA Interval Information Configuration Field Descriptions

Field	Description
ND router advertisements are sent every 60 to 100 seconds	ND RAs are sent at an interval randomly selected from a value between the minimum and maximum values. In this example, the minimum value is 60 seconds, and the maximum value is 100 seconds.

Field	Description
ND router advertisements are sent every 60 to 100 milliseconds	ND RAs are sent at an interval randomly selected from a value between the minimum and maximum values. In this example, the minimum value is 60 ms, and the maximum value is 100 ms.

Related Commands

Command	Description
ipv6 nd prefix	Configures which IPv6 prefixes are included in IPv6 router advertisements.
ipv6 nd ra interval	Configures the interval between IPv6 RA transmissions on an interface.
show ip interface	Displays the usability status of interfaces configured for IP.

show ipv6 mld snooping

To display Multicast Listener Discovery version 2 (MLDv2) snooping information, use the **show ipv6 mld snooping** command in privileged EXEC mode.

show ipv6 mld [**vrf** *vrf-name*] **snooping** {**explicit-tracking vlan** *vlan*| **mrouter** [**vlan** *vlan*]| **report-suppression vlan** *vlan*| **statistics vlan** *vlan*}

Syntax Description

vrf *vrf-name*	(Optional) Specifies a virtual routing and forwarding (VRF) configuration.
explicit-tracking vlan *vlan*	Displays the status of explicit host tracking.
mrouter	Displays the multicast router interfaces on an optional VLAN.
vlan *vlan*	(Optional) Specifies the VLAN number on the multicast router interfaces.
report-suppression vlan *vlan*	Displays the status of the report suppression.
statistics vlan *vlan*	Displays MLD snooping information on a VLAN.

Command Default

This command has no default settings.

Command Modes

Privileged EXEC

Command History

Release	Modification
12.2(18)SXE	This command was introduced on the Supervisor Engine 720.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
15.1(4)M	The **vrf** *vrf-name* keyword and argument were added.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

You can enter the **show ipv6 mld snooping mrouter** command without arguments to display all the multicast router interfaces.

Examples

This example shows how to display explicit tracking information on VLAN 25:

```
Router# show ipv6 mld snooping explicit-tracking vlan 25
Source/Group                    Interface    Reporter        Filter_mode
------------------------------------------------------------------------
10.1.1.1/226.2.2.2              Vl25:1/2     10.27.2.3       INCLUDE
10.2.2.2/226.2.2.2              Vl25:1/2     10.27.2.3       INCLUDE
```

This example shows how to display the multicast router interfaces in VLAN 1:

```
Router# show
ipv6 mld snooping mrouter vlan 1
vlan            ports
-----+----------------------------------------
  1          Gi1/1,Gi2/1,Fa3/48,Router
```

This example shows the MLD snooping statistics information for VLAN 25:

```
Router# show ipv6 mld
 snooping statistics interface vlan 25
Snooping staticstics for Vlan25
#channels:2
#hosts   :1

Source/Group            Interface       Reporter      Uptime        Last-Join   Last-Leave
10.1.1.1/226.2.2.2      Gi1/2:Vl25      10.27.2.3     00:01:47      00:00:50      -
10.2.2.2/226.2.2.2      Gi1/2:Vl25      10.27.2.3     00:01:47      00:00:50      -
```

Related Commands

Command	Description
ipv6 mld snooping	Enables MLDv2 snooping globally.
ipv6 mld snooping explicit-tracking	Enables explicit host tracking.
ipv6 mld snooping querier	Enables the MLDv2 snooping querier.
ipv6 mld snooping report-suppression	Enables report suppression on a VLAN.

show ipv6 nd ra-throttle policy

To display information about an IPv6 router advertisement (RA) throttler policy, use the **show ipv6 nd ra-throttle policy** command in privileged EXEC mode.

show ipv6 nd ra-throttle policy *policy-name*

Syntax Description

policy-name	RA throttler policy name.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

Use the **show ipv6 nd ra-throttle policy** to display IPv6 RA throttler information for troubleshooting purposes.

Examples

```
Device# show ipv6 nd ra-throttle policy policy2

Policy policy2 configuration:
        The throttle period will be coalesced and default to 600 seconds
        Applied to a port, this policy indicates a wired interface
        The maximum number of unthrottled RAs is configured on the vlan and defaults to 10
        The min and max numbers of unthrottled RAs per device will be coalesced and default
 to 10
       The behaviour upon RAs with an RFC 3775 interval option will be coalesced and default
 to passthrough

Policy applied on the following interfaces:
  Et0/0                vlan all
Policy applied on the following vlans:
  10,12-17
```

show ipv6 nd ra-throttle vlan

To display information about the actions of an IPv6 router advertisement (RA) throttler policy on a VLAN, use the **show ipv6 nd ra-throttle vlan** command in privileged EXEC mode.

show ipv6 nd ra-throttle vlan *vlan-id*[**advertising-routers| pending-hosts**]

Syntax Description

vlan-id	A VLAN or a collection of VLANs.
advertising-routers	(Optional) Displays information about devices that issued RAs recently.
pending-hosts	(Optional) Displays information about wireless hosts that are expecting RAs.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

Use the **show ipv6 nd ra-throttle vlan** command to display information about the actions of an IPv6 RA throttler policy on a VLAN.

Examples

```
Device# show ipv6 nd ra-throttle vlan vlan1

general information for vlan1
----------------------------------

 RAs             last period     this period    overall
 passed_through  1               1              2
 throttled       4               2              6

 no pending host

current policy is tutu coalesced as:

  throttle-period 90 seconds remaining 48
  max-through 0
  allow at-least 1 at-most 1
  interval-option passthrough
```

show ipv6 nd raguard policy

To display a router advertisements (RAs) guard policy on all interfaces configured with the RA guard feature, use the **show ipv6 nd raguard policy** command in privileged EXEC mode.

show ipv6 nd raguard policy [*policy-name*]

Syntax Description

policy-name	(Optional) RA guard policy name.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.2(4)S	This command was integrated into Cisco IOS Release 15.2(4)S.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **show ipv6 nd raguard policy** command displays the options configured for the policy on all interfaces configured with the RA guard feature.

Examples

The following example shows the policy configuration for a policy named raguard1 and all the interfaces where the policy is applied:

```
Router# show ipv6 nd raguard policy interface raguard1

Policy raguard1 configuration:
  device-role host
Policy applied on the following interfaces:
  Et0/0        vlan all
  Et1/0        vlan all
```

The table below describes the significant fields shown in the display.

Table 5: show ipv6 nd raguard policy Field Descriptions

Field	Description
Policy raguard1 configuration:	Configuration of the specified policy.

Field	Description
device-role host	The role of the device attached to the port. This device configuration is that of host.
Policy applied on the following interfaces:	The specified interface on which the RA guard feature is configured.

show ipv6 neighbor binding

To display contents of a binding table, use the **show ipv6 neighbor binding** command in privileged EXEC mode.

show ipv6 neighbor binding [**vlan** *vlan-id*| **interface** *type number*| **ipv6** *ipv6-address*| **mac** *mac-address*]

Syntax Description

vlan *vlan-id*	(Optional) Displays the binding table entries that match the specified VLAN.
interface *type number*	(Optional) Displays the binding table entries that match the specified interface type and number.
ipv6 *ipv6-address*	(Optional) Displays the binding table entries that match the specified IPv6 address.
mac *mac-address*	(Optional) Displays the binding table entries that match the specified Media Access Control (MAC) address.

Command Modes

Privileged EXEC (#)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE.	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **show ipv6 neighbor binding** command displays the contents of the binding table. The display output can be specified by the specified VLAN, interface, IPv6 address, or MAC address. If no keywords or arguments are entered, all binding table contents are displayed.

The following keyword and argument combinations are allowed:

- **vlan** *vlan-id*: Displays all entries for the specified VLAN.
- **interface** *type number*: Displays all entries for the specified interface.
- **ipv6** *ipv6-address* + **interface** *type number* + **vlan** *vlan-id*: Displays a single entry that matches these three keyword and argument combinations.

- **ipv6** *ipv6-address* + **interface** *type number*: Displays all entries for the specified IPv6 address and interface.
- **ipv6** *ipv6-address*: Displays all entries for the specified IPv6 address.

Examples

The following example displays the contents of a binding table:

```
Router# show ipv6 neighbor binding

address DB has 4 entries
Codes: L - Local, S - Static, ND - Neighbor Discovery
Preflevel (prlvl) values:
1:Not secure          2:MAC and LLA match   3:Cga authenticated
4:Dhcp assigned       5:Cert authenticated  6:Cga and Cert auth
7:Trusted port        8:Statically assigned
    IPv6 address            Link-Layer addr Interface   vlan  prlvl age state    Time left
ND  FE80::A8BB:CCFF:FE01:F500  AABB.CC01.F500  Et0/0     100  0002    0 REACHABLE  8850
L   FE80::21D:71FF:FE99:4900   001D.7199.4900  Vl100     100  0080 7203 DOWN       N/A
ND  2001:600::1               AABB.CC01.F500  Et0/0     100  0003    0 REACHABLE  3181
ND  2001:300::1               AABB.CC01.F500  Et0/0     100  0007    0 REACHABLE  9559
ND  2001:100::2               AABB.CC01.F600  Et1/0     200  0002    0 REACHABLE  9196
L   2001:400::1               001D.7199.4900  Vl100     100  0080 7188 DOWN       N/A
S   2001:500::1               000A.000B.000C  Fa4/13    300  0080 8676 STALE      N/A
```

The table below describes the significant fields shown in the display.

Table 6: show ipv6 neighbor binding Field Descriptions

Field	Description
address DB has *n* entries	Number of entries in the specified database.

Related Commands

Command	Description
ipv6 neighbor binding	Changes the defaults of neighbor binding entries in a binding table.

show ipv6 neighbors

To display IPv6 neighbor discovery (ND) cache information, use the **show ipv6 neighbors** command in user EXEC or privileged EXEC mode.

show ipv6 neighbors [*interface-type interface-number*| *ipv6-address*| *ipv6-hostname*| **statistics**]

Syntax Description

interface-type	(Optional) Specifies the type of the interface from which IPv6 neighbor information is to be displayed.
interface-number	(Optional) Specifies the number of the interface from which IPv6 neighbor information is to be displayed.
ipv6-address	(Optional) Specifies the IPv6 address of the neighbor. This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.
ipv6-hostname	(Optional) Specifies the IPv6 hostname of the remote networking device.
statistics	(Optional) Displays ND cache statistics.

Command Default

All IPv6 ND cache entries are listed.

Command Modes

User EXEC (>) Privileged EXEC (#)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.2(8)T	This command was modified. Support for static entries in the IPv6 neighbor discovery cache was added to the command output.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.

Release	Modification
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1 and introduced on Cisco ASR 1000 Series devices.
Cisco IOS XE Release 2.6	This command was modified. This command was updated to display the number and the limit of ND cache entries on a particular interface.
15.1(3)T	This command was integrated into Cisco IOS Release 15.1(3)T.
15.2(2)SNG	This command was implemented on the Cisco ASR 901 Series Aggregation Services devices.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

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.

Specifying the **statistics** keyword displays ND cache statistics.

Examples

The following is sample output from the **show ipv6 neighbors** command when entered with an interface type and number:

```
Device# show ipv6 neighbors ethernet 2
IPv6 Address                              Age Link-layer Addr State Interface
2000:0:0:4::2                               0 0003.a0d6.141e  REACH Ethernet2
FE80::203:A0FF:FED6:141E                    0 0003.a0d6.141e  REACH Ethernet2
3001:1::45a                                 - 0002.7d1a.9472  REACH Ethernet2
```

The following is sample output from the **show ipv6 neighbors** command when entered with an IPv6 address:

```
Device# 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 Ethernet2
```

The table below describes the significant fields shown in the displays.

Table 7: show ipv6 neighbors 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.

Field	Description
Link-layer Addr	MAC address. If the address is unknown, a hyphen (-) is displayed.

Field	Description
State	

Field	Description
	The state of the neighbor cache entry. Following are the states for dynamic entries in the IPv6 neighbor discovery cache: • 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. • 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. • 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. • 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. • PROBE--A reachability confirmation is actively sought by resending neighbor solicitation messages every RetransTimer milliseconds until a reachability confirmation is received. • ????--Unknown state. Following are the possible states for static entries in the IPv6 neighbor discovery cache: • INCMP (Incomplete)--The interface for this entry is down. • REACH (Reachable)--The interface for this entry is up.

Field	Description
	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 was reachable.

The following is sample output from the **show ipv6 neighbors** command with the **statistics** keyword:

```
Device# show ipv6 neighbor statistics

IPv6 ND Statistics
 Entries 2, High-water 2, Gleaned 1, Scavenged 0
 Entry States
   INCMP 0  REACH 0  STALE 2  GLEAN 0  DELAY 0  PROBE 0
 Resolutions (INCMP)
   Requested 1, timeouts 0, resolved 1, failed 0
   In-progress 0, High-water 1, Throttled 0, Data discards 0
 Resolutions (PROBE)
   Requested 3, timeouts 0, resolved 3, failed 0
```

The table below describes the significant fields shown in this display:

Table 8: show ipv6 neighbors statistics Field Descriptions

Field	Description
Entries	Total number of ND neighbor entries in the ND cache.
High-Water	Maximum amount (so far) of ND neighbor entries in ND cache.
Gleaned	Number of ND neighbor entries gleaned (that is, learned from a neighbor NA or other ND packet).
Scavenged	Number of stale ND neighbor entries that have timed out and been removed from the cache.
Entry States	Number of ND neighbor entries in each state.

Field	Description
Resolutions (INCMP)	Statistics for neighbor resolutions attempted in INCMP state (that is, resolutions prompted by a data packet). Details about the resolutions attempted in INCMP state are follows: • Requested--Total number of resolutions requested. • Timeouts--Number of timeouts during resolutions. • Resolved--Number of successful resolutions. • Failed--Number of unsuccessful resolutions. • In-progress--Number of resolutions in progress. • High-water--Maximum number (so far) of resolutions in progress. • Throttled--Number of times resolution request was ignored due to maximum number of resolutions in progress limit. • Data discards--Number of data packets discarded that are awaiting neighbor resolution.
Resolutions (PROBE)	Statistics for neighbor resolutions attempted in PROBE state (that is, re-resolutions of existing entries prompted by a data packet): • Requested--Total number of resolutions requested. • Timeouts--Number of timeouts during resolutions. • Resolved--Number of successful resolutions. • Failed--Number of unsuccessful resolutions.

show ipv6 protocols

To display the parameters and the current state of the active IPv6 routing protocol processes, use the **show ipv6 protocols** command in user EXEC or privileged EXEC mode.

show ipv6 protocols [summary]

Syntax Description

summary	(Optional) Displays the configured routing protocol process names.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
12.2(8)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.4(15)T	This command was modified. The command output was enhanced to provide Enhanced Interior Gateway Routing Protocol (EIGRP) information, including the vector metric.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.4	This command was implemented on Cisco ASR 1000 Series Aggregation Services Routers.
Cisco IOS XE Release 3.6	This command was modified. The command output was enhanced to include information about EIGRP IPv6 Nonstop Forwarding (NSF).
15.2(2)S	This command was modified. The command output was enhanced to include information about EIGRP IPv6 NSF.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The information displayed by the **show ipv6 protocols** command is useful in debugging routing operations.

Examples

The following sample output from the **show ipv6 protocols** command displays Intermediate System-to-Intermediate System (IS-IS) routing protocol information:

```
Device# show ipv6 protocols

IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "static"
IPv6 Routing Protocol is "isis"
  Interfaces:
    Ethernet0/0/3
    Ethernet0/0/1
    Serial1/0/1
    Loopback1 (Passive)
    Loopback2 (Passive)
    Loopback3 (Passive)
    Loopback4 (Passive)
    Loopback5 (Passive)
  Redistribution:
    Redistributing protocol static at level 1
  Inter-area redistribution
    Redistributing L1 into L2 using prefix-list word
  Address Summarization:
    L2: 33::/16  advertised with metric 0
    L2: 44::/16  advertised with metric 20
    L2: 66::/16  advertised with metric 10
    L2: 77::/16  advertised with metric 10
```

The table below describes the significant fields shown in the display.

Table 9: show ipv6 protocols Field Descriptions for IS-IS Processes

Field	Description
IPv6 Routing Protocol is	Specifies the IPv6 routing protocol used.
Interfaces	Specifies the interfaces on which the IPv6 IS-IS protocol is configured.
Redistribution	Lists the protocol that is being redistributed.
Inter-area redistribution	Lists the IS-IS levels that are being redistributed into other levels.
using prefix-list	Names the prefix list used in the interarea redistribution.
Address Summarization	Lists all the summary prefixes. If the summary prefix is being advertised, "advertised with metric *x*" will be displayed after the prefix.

The following sample output from the **show ipv6 protocols** command displays the Border Gateway Protocol (BGP) information for autonomous system 30:

```
Device# show ipv6 protocols

IPv6 Routing Protocol is "bgp 30"
  IGP synchronization is disabled
  Redistribution:
    Redistributing protocol connected
  Neighbor(s):
    Address                    FiltIn FiltOut Weight RoutemapIn RoutemapOut
    2001:DB8:0:ABCD::1            5       7    200
    2001:DB8:0:ABCD::2                               rmap-in    rmap-out
    2001:DB8:0:ABCD::3                               rmap-in    rmap-out
```

The table below describes the significant fields shown in the display.

Table 10: show ipv6 protocols Field Descriptions for BGP Process

Field	Description
IPv6 Routing Protocol is	Specifies the IPv6 routing protocol used.
Redistribution	Lists the protocol that is being redistributed.
Address	Neighbor IPv6 address.
FiltIn	AS-path filter list applied to input.
FiltOut	AS-path filter list applied to output.
Weight	Neighbor weight value used in BGP best path selection.
RoutemapIn	Neighbor route map applied to input.
RoutemapOut	Neighbor route map applied to output.

The following is sample output from the **show ipv6 protocols summary** command:

```
Device# show ipv6 protocols summary

Index Process Name
0     connected
1     static
2     rip myrip
3     bgp 30
```

The following sample output from the **show ipv6 protocols** command displays the EIGRP information including the vector metric and EIGRP IPv6 NSF:

```
Device# show ipv6 protocols

IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "bgp 1"
  IGP synchronization is disabled
  Redistribution:
    None
IPv6 Routing Protocol is "bgp multicast"
```

```
IPv6 Routing Protocol is "ND"
IPv6 Routing Protocol is "eigrp 1"
EIGRP-IPv6 VR(name) Address-Family Protocol for AS(1)
  Metric weight K1=1, K2=0, K3=1, K4=0, K5=0 K6=0
  Metric rib-scale 128
  Metric version 64bit
  NSF-aware route hold timer is 260
  EIGRP NSF enabled
     NSF signal timer is 15s
     NSF converge timer is 65s
  Router-ID: 10.1.2.2
  Topology : 0 (base)
    Active Timer: 3 min
    Distance: internal 90 external 170
    Maximum path: 16
    Maximum hopcount 100
    Maximum metric variance 1
    Total Prefix Count: 0
    Total Redist Count: 0

  Interfaces:
  Redistribution:
    None
```

The following example displays IPv6 protocol information after configuring redistribution in an Open Shortest Path First (OSPF) domain:

```
Device# redistribute ospf 1 match internal
Device(config-rtr)# end
Device# show ipv6 protocols

IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "ND"
IPv6 Routing Protocol is "rip 1"
  Interfaces:
    Ethernet0/1
    Loopback9
  Redistribution:
    Redistributing protocol ospf 1 (internal)
IPv6 Routing Protocol is "ospf 1"
  Interfaces (Area 0):
    Ethernet0/0
  Redistribution:
    None
```

show ipv6 route

To display contents of the IPv6 routing table, use the **show ipv6 route** command in user EXEC or privileged EXEC mode.

show ipv6 route [*ipv6-address*| *ipv6-prefix*/*prefix-length* [**longer-prefixes**]| [*protocol*] | [**repair**] | [**updated** [**boot-up**] [*day month*] [*time*]]| **interface** *type number*| **nd**| **nsf**| **table** *table-id* **| watch**]

Syntax Description

ipv6-address	(Optional) Displays routing information for a specific IPv6 address.
ipv6-prefix	(Optional) Displays routing information for a specific IPv6 network.
/*prefix-length*	(Optional) 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.
longer-prefixes	(Optional) Displays output for longer prefix entries.
protocol	(Optional) The name of a routing protocol or the keyword **connected**, **local**, **mobile**, or **static**. If you specify a routing protocol, use one of the following keywords: **bgp**, **isis**, **eigrp**, **ospf**, or **rip**.
repair	(Optional) Displays routes with repair paths.
updated	(Optional) Displays routes with time stamps.
boot-up	(Optional) Displays routing information since bootup.
day month	(Optional) Displays routes since the specified day and month.
time	(Optional) Displays routes since the specified time, in *hh:mm* format.
interface	(Optional) Displays information about the interface.
type	(Optional) Interface type.
number	(Optional) Interface number.
nd	(Optional) Displays only routes from the IPv6 Routing Information Base (RIB) that are owned by Neighbor Discovery (ND).

nsf	(Optional) Displays routes in the nonstop forwarding (NSF) state.
repair	(Optional)
table *table-id*	(Optional) Displays IPv6 RIB table information for the specified table ID. The table ID must be in hexadecimal format. The range is from 0 to 0-0xFFFFFFFF.
watch	(Optional) Displays information about route watchers.

Command Default

If none of the optional syntax elements is chosen, all IPv6 routing information for all active routing tables is displayed.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.2(8)T	This command was modified. The **isis** keyword was added, and the I1 - ISIS L1, I2 - ISIS L2, and IA - ISIS interarea fields were included in the command output.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S. The timer information was removed, and an indicator was added to display IPv6 Multiprotocol Label Switching (MPLS) interfaces.
12.2(13)T	This command was modified. The timer information was removed, and an indicator was added to display IPv6 MPLS virtual interfaces.
12.2(14)S	This command was modified. The **longer-prefixes** keyword was added.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Release	Modification
Cisco IOS XE Release 2.1	This command was implemented on Cisco ASR 1000 Series Aggregation Services Routers.
12.4(24)T	This command was modified in a release earlier than Cisco IOS Release 12.4(24)T. The **table**, **nsf**, **watch**, and **updated** keywords and the *day, month, table-id*, and *time* arguments were added.
15.2(2)S	This command was modified. The command output was enhanced to include route tag values in dotted-decimal format.
Cisco IOS XE Release 3.6S	This command was modified. The command output was enhanced to include route tag values in dotted-decimal format.
15.1(1)SY	The **nd** keyword was added.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **show ipv6 route** command provides output similar to the **show ip route** command, except that the information is IPv6-specific.

When the *ipv6-address* or *ipv6-prefix*/*prefix-length* argument is specified, the longest match lookup is performed from the routing table, and only route information for that address or network is displayed. When a routing protocol is specified, only routes for that protocol are displayed. When the **connected**, **local**, **mobile**, or **static** keyword is specified, only the specified type of route is displayed. When the **interface** keyword and *type* and *number* arguments are specified, only the specified interface-specific routes are displayed.

Examples

The following is sample output from the **show ipv6 route** command when no keywords or arguments are specified:

```
Device# show ipv6 route

IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       I1 - ISIS L1, I2 - ISIS L2, IA - IIS interarea
B   2001:DB8:4::2/48 [20/0]
     via FE80::A8BB:CCFF:FE02:8B00, Serial6/0
L   2001:DB8:4::3/48 [0/0]
     via ::, Ethernet1/0
C   2001:DB8:4::4/48 [0/0]
     via ::, Ethernet1/0
LC  2001:DB8:4::5/48 [0/0]
     via ::, Loopback0
L   2001:DB8:4::6/48 [0/0]
     via ::, Serial6/0
C   2001:DB8:4::7/48 [0/0]
     via ::, Serial6/0
S   2001:DB8:4::8/48 [1/0]
     via 2001:DB8:1::1, Null
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
```

The table below describes the significant fields shown in the display.

Table 11: show ipv6 route Field Descriptions

Field	Description
Codes:	Indicates the protocol that derived the route. Values are as follows: • B—BGP derived • C—Connected • I1—ISIS L1—Integrated IS-IS Level 1 derived • I2—ISIS L2—Integrated IS-IS Level 2 derived • IA—ISIS interarea—Integrated IS-IS interarea derived • L—Local • R—RIP derived • S—Static
2001:DB8:4::2/48	Indicates the IPv6 prefix of the remote network.
[20/0]	The first number in brackets is the administrative distance of the information source; the second number is the metric for the route.
via FE80::A8BB:CCFF:FE02:8B00	Specifies the address of the next device to the remote network.

When the *ipv6-address* or *ipv6-prefix/prefix-length* argument is specified, only route information for that address or network is displayed. The following is sample output from the **show ipv6 route** command when IPv6 prefix 2001:DB8::/35 is specified. The fields in the display are self-explanatory.

```
Device# show ipv6 route 2001:DB8::/35

IPv6 Routing Table - 261 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
B 2001:DB8::/35 [20/3]
  via FE80::60:5C59:9E00:16, Tunnel1
```

When you specify a protocol, only routes for that particular routing protocol are shown. The following is sample output from the **show ipv6 route bgp** command. The fields in the display are self-explanatory.

```
Device# show ipv6 route bgp

IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
B   2001:DB8:4::4/64 [20/0]
     via FE80::A8BB:CCFF:FE02:8B00, Serial6/0
```

The following is sample output from the **show ipv6 route local** command. The fields in the display are self-explanatory.

```
Device# show ipv6 route local

IPv6 Routing Table - 9 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea
L   2001:DB8:4::2/128 [0/0]
     via ::, Ethernet1/0
LC  2001:DB8:4::1/128 [0/0]
     via ::, Loopback0
L   2001:DB8:4::3/128 [0/0]
     via ::, Serial6/0
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
```

The following is sample output from the **show ipv6 route** command when the 6PE multipath feature is enabled. The fields in the display are self-explanatory.

```
Device# show ipv6 route

IPv6 Routing Table - default - 19 entries
Codes:C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
.
.
.
B   2001:DB8::/64 [200/0]
     via ::FFFF:172.11.11.1
     via ::FFFF:172.30.30.1
```

Related Commands

Command	Description
ipv6 route	Establishes a static IPv6 route.
show ipv6 interface	Displays IPv6 interface information.
show ipv6 route summary	Displays the current contents of the IPv6 routing table in summary format.
show ipv6 tunnel	Displays IPv6 tunnel information.

show ipv6 snooping capture-policy

To display message capture policies, use the **show ipv6 snooping capture-policy** command in user EXEC or privileged EXEC mode.

show ipv6 snooping capture-policy [**interface** *type number*]

Syntax Description

interface *type number*	(Optional) Displays first-hop message types on the specified interface type and number.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **show ipv6 snooping capture-policy** command displays IPv6 first-hop message capture policies.

Examples

The following example shows **show ipv6 snooping capture-policy** command output on the Ethernet 0/0 interface, on which the IPv6 Neighbor Discovery Protocol (NDP) Inspection and Router Advertisement (RA) Guard features are configured:

```
Router# show ipv6 snooping capture-policy

Hardware policy registered on Et0/0
Protocol  Protocol value  Message  Value  Action  Feature
ICMP      58              RS       85     punt    RA Guard
                                          punt    ND Inspection
ICMP      58              RA       86     drop    RA guard
                                          punt    ND Inspection
ICMP      58              NS       87     punt    ND Inspection
ICMP      58              NA       88     punt    ND Inspection
ICMP      58              REDIR    89     drop    RA Guard
                                          punt    ND Inspection
```

The table below describes the significant fields shown in the display.

Table 12: show ipv6 snooping capture-policy Field Descriptions

Field	Description
Hardware policy registered on Fa4/11	A hardware policy contains a programmatic access list (ACL), with a list of access control entries (ACEs).
Protocol	The protocol whose packets are being inspected.
Message	The type of message being inspected.
Action	Action to be taken on the packet.
Feature	The inspection feature for this information.

show ipv6 snooping counters

To display information about the packets counted by the interface counter, use the **show ipv6 snooping counters**command in user EXEC or privileged EXEC mode.

show ipv6 snooping counters [**interface** *type number*]

Syntax Description

interface *type number*	(Optional) Displays first hop packets that match the specified interface type and number.

Command Modes

User EXEC Privileged EXEC (#)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **show ipv6 snooping counters** command shows packets handled by the switcher that are being counted in interface counters. The switcher counts packets captured per interface and records whether the packet was received, sent, or dropped. If a packet is dropped, the reason for the drop and the feature that caused the drop are both also provided.

Examples

The following examples shows information about packets counted on interface FastEthernet4/12:

```
Router# show ipv6 snooping counters interface Fa4/12
Received messages on Fa4/12:
Protocol        Protocol message
ICMPv6          RS      RA      NS      NA      REDIR   CPS     CPA
                0       4256    0       0       0       0       0
Bridged messages from Fa4/12:
Protocol        Protocol message
ICMPv6          RS      RA      NS      NA      REDIR   CPS     CPA
                0       4240    0       0       0       0       0
Dropped messages on Fa4/12:
Feature/Message RS      RA      NS      NA      REDIR   CPS     CPA
RA guard        0       16      0       0       0       0       0
Dropped reasons on Fa4/12:
RA guard         16   RA drop - reason:RA/REDIR received on un-authorized port
```

The table below describes the significant fields shown in the display.

Table 13: show ipv6 snooping counters Field Descriptions

Field	Description
Received messages on Fa4/12:	The messages received on an interface.
Protocol	The protocol for which messages are being counted.
Protocol message	The type of protocol messages being counted.
Bridged messages from Fa4/12:	Bridged messages from the interface.
Dropped messages an Fa4/12:	The messages dropped on the interface.
Feature/message	The feature that caused the drop, and the type and number of messages dropped.
RA drop - reason:RA/REDIR received on un-authorized port	The reason these messages were dropped.

show ipv6 snooping features

To display information about about snooping features configured on the router, use the **show ipv6 snooping features** command in user EXEC or privileged EXEC mode.

show ipv6 snooping features

Syntax Description

This command has no arguments or keywords.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.

Usage Guidelines

The **show ipv6 snooping features** command shows the first hop features that are configured on the router.

Examples

The following example shows that both IPv6 ND inspection and IPv6 RA Guard are configured on the router:

```
Router# show ipv6 snooping features

Feature name   priority state
RA guard          100   READY
NDP inspection     20   READY
```

The table below describes the significant fields shown in the display.

Table 14: show ipv6 snooping features Field Descriptions

Field	Description
Feature name	The names of the IPv6 global policy features configured on the router.
Priority	The priority of the specified feature.
State	The state of the specified feature.

show ipv6 snooping policies

To display information about the configured policies and the interfaces to which they are attached, use the **show ipv6 snooping policies** command in user EXEC or privileged EXEC mode.

show ipv6 snooping policies [**interface** *type number*]

Syntax Description

interface *type number*	(Optional) Displays policies that match the specified interface type and number.

Command Modes

User EXEC (>)

Privileged EXEC (#)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.

Usage Guidelines

The **show ipv6 snooping policies** command displaying all policies that are configured and lists the interfaces to which they are attached.

Examples

The following examples shows information about all policies configured:

```
Device# show ipv6 snooping policies

NDP inspection policies configured:
Policy      Interface    Vlan
------      ---------    ----
trusted      Et0/0        all
             Et1/0        all
untrusted    Et2/0        all
RA guard policies configured:
Policy      Interface    Vlan
------      ---------    ----
host         Et0/0        all
             Et1/0        all
router       Et2/0        all
```

The table below describes the significant fields shown in the display.

Table 15: show ipv6 first-hop policies Field Descriptions

Field	Description
NDP inspection policies configured:	Description of the policies configured for a specific feature.
Policy	Whether the policy is trusted or untrusted.
Interface	The interface to which a policy is attached.

show ipv6 traffic

To display statistics about IPv6 traffic, use the **show ipv6 traffic**command in user EXEC or privileged EXEC mode.

show ipv6 traffic [**interface** [*interface type number*]]

Syntax Description

interface	(Optional) All interfaces. IPv6 forwarding statistics for all interfaces on which IPv6 forwarding statistics are being kept will be displayed.
interface type number	(Optional) Specified interface. Interface statistics that have occurred since the statistics were last cleared on the specific interface are displayed.

Command Modes

User EXEC Privileged EXEC

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S, and output fields were added.
12.2(13)T	The modification to add output fields was integrated into this release.
12.2(14)S	This command was integrated into Cisco IOS Release 12.2(14)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
12.2(33)SRC	The *interface* argument and **interface** keyword were added.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SB.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Series devices.

Release	Modification
15.2(2)SNG	This command was implemented on the Cisco ASR 901 Series Aggregation Services devices.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The **show ipv6 traffic**command provides output similar to the **show ip traffic**command, except that it is IPv6-specific.

Examples

The following is sample output from the **show ipv6 traffic**command:

```
Device# 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 not a device
         0 fragments, 0 total reassembled
         0 reassembly timeouts, 0 reassembly failures
         0 unicast RPF drop, 0 suppressed RPF drop
  Sent:  0 generated, 0 forwarded
         0 fragmented into 0 fragments, 0 failed
         0 encapsulation 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 info type, 0 unknown error type
        unreach: 0 routing, 0 admin, 0 neighbor, 0 address, 0 port
        parameter: 0 error, 0 header, 0 option
        0 hopcount expired, 0 reassembly timeout,0 too big
        0 echo request, 0 echo reply
        0 group query, 0 group report, 0 group reduce
        0 device solicit, 0 device advert, 0 redirects
```

The following is sample output for the show ipv6 interface command without IPv6 CEF running:

```
Device# show ipv6 interface ethernet 0/1/1
Ethernet0/1/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::203:FDFF:FE49:9
  Description: sat-2900a f0/12
  Global unicast address(es):
    7::7, subnet is 7::/32
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:7
    FF02::1:FF49:9
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  Input features: RPF
  Unicast RPF access-list MINI
    Process Switching:
      0 verification drops
      0 suppressed verification drops
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
```

The following is sample output for the show ipv6 interface command with IPv6 CEF running:

```
Device# show ipv6 interface ethernet 0/1/1
Ethernet0/1/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::203:FDFF:FE49:9
  Description: sat-2900a f0/12
  Global unicast address(es):
    7::7, subnet is 7::/32
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:7
    FF02::1:FF49:9
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  Input features: RPF
  Unicast RPF access-list MINI
    Process Switching:
      0 verification drops
      0 suppressed verification drops
    CEF Switching:
      0 verification drops
      0 suppressed verification drops
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
  ND advertised 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.
```

The table below describes the significant fields shown in the display.

Table 16: show ipv6 traffic Field Descriptions

Field	Description
source-routed	Number of source-routed packets.
truncated	Number of truncated packets.
format errors	Errors that can result from checks performed on header fields, the version number, and packet length.
not a device	Message sent when IPv6 unicast routing is not enabled.
0 unicast RPF drop, 0 suppressed RPF drop	Number of unicast and suppressed reverse path forwarding (RPF) drops.
failed	Number of failed fragment transmissions.
encapsulation failed	Failure that can result from an unresolved address or try-and-queue packet.
no route	Counted when the software discards a datagram it did not know how to route.

Field	Description
unreach	Unreachable messages received are as follows: • routing--Indicates no route to the destination. • admin--Indicates that communication with the destination is administratively prohibited. • neighbor--Indicates that the destination is beyond the scope of the source address. For example, the source may be a local site or the destination may not have a route back to the source. • address--Indicates that the address is unreachable. • port--Indicates that the port is unreachable.
Unicast RPF access-list MINI	Unicast RPF access-list in use.
Process Switching	Displays process RPF counts, such as verification and suppressed verification drops.
CEF Switching	Displays CEF switching counts, such as verification drops and suppressed verification drops.

summary-prefix (OSPFv3)

To configure an IPv6 summary prefix in Open Shortest Path First version 3 (OSPFv3), use the **summary-prefix** command in OSPFv3 router configuration mode, IPv6 address family configuration mode, or IPv4 address family configuration mode. To restore the default, use the **no** form of this command.

summary-prefix *prefix* [**not-advertise**| **tag** *tag-value*] [**nssa-only**]

no summary-prefix *prefix* [**not-advertise**| **tag** *tag-value*] [**nssa-only**]

Syntax Description

prefix	IPv6 route prefix for the destination.
not-advertise	(Optional) Suppresses routes that match the specified prefix and mask pair. This keyword applies to OSPFv3 only.
tag *tag-value*	(Optional) Specifies the tag value that can be used as a match value for controlling redistribution via route maps. This keyword applies to OSPFv3 only.
nssa-only	(Optional) Limits the scope of the prefix to the area. Sets the nssa-only attribute for the summary route (if any) generated for the specified prefix.

Command Default

No IPv6 summary prefix is defined.

Command Modes

OSPFv3 router configuration mode (config-router)

IPv6 address family configuration (config-router-af)

IPv4 address family configuration (config-router-af)

Command History

Release	Modification
12.0(24)S	This command was introduced.
12.2(15)T	This command was integrated into Cisco IOS Release 12.2(15)T.
12.2(18)S	This command was integrated into Cisco IOS Release 12.2(18)S.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.

Release	Modification
15.1(3)S	This command was modified. The command can be enabled in an IPv4 or IPv6 OSPFv3 process.
Cisco IOS XE Release 3.4S	This command was modified. The command can be enabled in an IPv4 or IPv6 OSPFv3 process.
15.2(1)T	This command was modified. The command can be enabled in an IPv4 or IPv6 OSPFv3 process.
15.2(4)S	This command was modified. The **nssa-only** keyword was added.
15.1(1)SY	This command was modified. The command can be enabled in an IPv4 or IPv6 OSPFv3 process.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The summary-prefix command can be used to summarize devices redistributed from other routing protocols. Multiple groups of addresses can be summarized. The metric used to advertise the summary is the smallest metric of all the more specific routes. This command helps reduce the size of the routing table.

Specify the **nssa-only** keyword to clear the propagate bit (P-bit) when external routes are redistributed into a not-so-stubby area (NSSA). Doing so prevents corresponding NSSA external link state advertisements (LSAs) from being translated into other areas.

Examples

In the following example, the summary prefix 2051:0:0:10::/60 includes addresses beginning at 2051:0:0:10::/60 up to (but not including) 2051:0:0:20::/128. Only the address 2051:0:0:10::/60 is advertised in an external LSA:

```
summary-prefix 2051:0:0:10::/60
```

Related Commands

router ospfv3	Enables OSPFv3 router configuration mode for the IPv4 or IPv6 address family.

throttle-period

To configure the throttle period in an IPv6 router advertisement (RA) throttler policy, use the **throttle-period** command in IPv6 RA throttle policy configuration mode. To reset this command to its default, use the **no** form of the command.

throttle-period { **inherit**| *seconds*}

Syntax Description

inherit	The throttle period setting is inherited from target policies.
seconds	Duration of the throttle period, in seconds. The range is from 10 through 86,400 seconds.

Command Default

600 seconds (10 minutes)

Command Modes

IPv6 RA throttle policy configuration

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

The **throttle-period** command is only valid for policies attached to a VLAN or VLANs. If you try to configure this command on a port, the port ignores it.

Examples

```
Device(config)# ipv6 nd ra-throttle policy policy1
Device(config-nd-ra-throttle)# throttle-period 300
```

timers spf (IPv6)

To turn on Open Shortest Path First (OSPF) for IPv6 shortest path first (SPF) throttling, use the **timers spf** command in router configuration mode. To turn off SPF throttling, use the **no** form of this command.

timers spf *delay holdtime*

no timers spf

Syntax Description

delay	Delay (in milliseconds) in receiving a change in the SPF calculation. The range is from 0 through 4294967295. The default is 5 milliseconds.
holdtime	Hold time (in milliseconds) between consecutive SPF calculations. The range is from 0 through 4294967295. The default is 10 milliseconds.

Command Default

OSPF for IPv6 throttling is always enabled.

Command Modes

Router configuration

Command History

Release	Modification
12.2(15)T	This command was introduced.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The first wait interval between SPF calculations is the amount of time in milliseconds specified by the *delay* argument. Each consecutive wait interval is two times the current hold level in milliseconds until the wait time reaches the maximum time in milliseconds as specified by the *holdtime* argument. Subsequent wait times remain at the maximum until the values are reset or a link-state advertisement (LSA) is received between SPF calculations.

Examples

The following example shows a router configured with the delay and hold-time interval values for the **timers spf** command set at 40 and 50 milliseconds, respectively.

```
Router(config)# ipv6 router ospf 1
Router(config-router)# timers spf 40 50
```

Related Commands

Command	Description
show ipv6 ospf	Displays general information about OSPF for IPv6 routing processes.

timers throttle lsa

To set rate-limiting values for Open Shortest Path First (OSPF) for IPv6 link-state advertisement (LSA) generation, use the **timers throttle lsa**command in router configuration mode. To restore the default values, use the **no** form of this command.

timers throttle lsa *start-interval hold-interval max-interval*

no timers throttle lsa

Syntax Description

start-interval	Minimum delay in milliseconds for the generation of LSAs. The first instance of LSA is always generated immediately upon a local OSPF for IPv6 topology change. The generation of the next LSA is not before the start interval. The range is from 0 to 600,000 milliseconds. The default is 0 milliseconds, which means no delay; the LSA is sent immediately.
hold-interval	Incremental time in milliseconds. This value is used to calculate the subsequent rate limiting times for LSA generation. The range is from 1 to 600,000 milliseconds. The default value is 5000 milliseconds.
max-interval	Maximum wait time in milliseconds between generation of the same LSA. The range is from 1 to 600,000 milliseconds. The default value is 5000 milliseconds.

Command Default

start-interval : 0 milliseconds*hold-interval:*5000 milliseconds*max-interval*: 5000 milliseconds

Command Modes

OSPF for IPv6 router configuration (config-rtr) Router configuration (config-router)

Command History

Release	Modification
12.2(33)SRC	This command was introduced.
12.2(33)SB	This command was integrated into Cisco IOS Release 12.2(33)SRA.
Cisco IOS XE Release 2.1	This command was introduced on Cisco ASR 1000 Series Routers.
15.0(1)M	This command was integrated into Cisco IOS Release 12.5(1)M.
12.2(33)XNE	This command was modified. It was integrated into Cisco IOS Release 12.2(33)XNE.

Release	Modification
15.1(1)SY	This command was modified. It was integrated into Cisco IOS Release 15.1(1)SY.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

The "same LSA" is defined as an LSA instance that contains the same LSA ID number, LSA type, and advertising router ID. We suggest you keep the *milliseconds*value of the **timers lsa arrival**command less than or equal to the *hold-interval* value of the **timers throttle lsa**command.

Examples

This example customizes OSPF LSA throttling so that the start interval is 200 milliseconds, the hold interval is 10,000 milliseconds, and the maximum interval is 45,000 milliseconds. The minimum interval between instances of receiving the same LSA is 2000 milliseconds.

```
router ospf 1
 log-adjacency-changes
 timers throttle lsa 200 10000 45000
 timers lsa arrival 2000
 network 10.10.4.0 0.0.0.255 area 24
 network 10.10.24.0 0.0.0.255 area 24
```

This example customizes IPv6 OSPF LSA throttling so that the start interval is 500 milliseconds, the hold interval is 1,000 milliseconds, and the maximum interval is 10,000 milliseconds.

```
ipv6 router ospf 1
 log-adjacency-changes
 timers throttle lsa 500 1000 10000
```

Related Commands

Command	Description
show ipv6 ospf	Displays information about OSPF for IPv6 routing processes.
timers lsa arrival	Sets the minimum interval at which the software accepts the same LSA from OSPF neighbors.

tracking

To override the default tracking policy on a port, use the **tracking**command in Neighbor Discovery (ND) inspection policy configuration mode.

tracking {**enable** [**reachable-lifetime** {*value*| **infinite**}]| **disable** [**stale-lifetime** {*value*| **infinite**}]}

Syntax Description

enable	Tracking is enabled.
reachable-lifetime	(Optional) The maximum amount of time a reachable entry is considered to be directly or indirectly reachable without proof of reachability. • The **reachable-lifetime** keyword can be used only with the **enable** keyword. • Use of the **reachable-lifetime** keyword overrides the global reachable lifetime configured by the **ipv6 neighbor binding reachable-lifetime** command.
value	Lifetime value, in seconds. The range is from 1 to 86400, and the default is 300.
infinite	Keeps an entry in a reachable or stale state for an infinite amount of time.
disable	Disables tracking.
stale-lifetime	(Optional) Keeps the time entry in a stale state, which overwrites the global stale-lifetime configuration. • The stale lifetime is 86,400 seconds. • The **stale-lifetime** keyword can be used only with the **disable** keyword. • Use of the **stale-lifetime** keyword overrides the global stale lifetime configured by the **ipv6 neighbor binding stale-lifetime** command.

Command Default

The time entry is kept in a reachable state.

Command Modes

ND inspection policy configuration (config-nd-inspection)

Command History

Release	Modification
12.2(50)SY	This command was introduced.
15.0(2)SE	This command was integrated into Cisco IOS Release 15.0(2)SE.
15.3(1)S	This command was integrated into Cisco IOS Release 15.3(1)S.

Usage Guidelines

The **tracking** command overrides the default tracking policy set by the **ipv6 neighbor tracking** command on the port on which this policy applies. This function is useful on trusted ports where, for example, you may not want to track entries but want an entry to stay in the binding table to prevent it from being stolen.

The **reachable-lifetime** keyword is the maximum time an entry will be considered reachable without proof of reachability, either directly through tracking or indirectly through ND inspection. After the **reachable-lifetime** value is reached, the entry is moved to stale. Use of the **reachable-lifetime** keyword with the **tracking** command overrides the global reachable lifetime configured by the **ipv6 neighbor binding reachable-lifetime** command.

The **stale-lifetime** keyword is the maximum time an entry is kept in the table before it is deleted or the entry is proven to be reachable, either directly or indirectly. Use of the **stale-lifetime** keyword with the **tracking** command overrides the global stale lifetime configured by the **ipv6 neighbor binding stale-lifetime** command.

Examples

The following example defines an ND policy name as policy1, places the router in ND inspection policy configuration mode, and configures an entry to stay in the binding table for an infinite length of time on a trusted port:

```
Router(config)# ipv6 nd inspection policy policy1
Router(config-nd-inspection)# tracking disable stale-lifetime infinite
```

Related Commands

Command	Description
ipv6 nd inspection policy	Defines the ND inspection policy name and enters ND inspection policy configuration mode.
ipv6 neighbor binding	Changes the defaults of neighbor binding entries in a binding table.
ipv6 neighbor tracking	Enables tracking of entries in the binding table.
ipv6 nd raguard policy	Defines the RA guard policy name and enters RA guard policy configuration mode.

tunnel mode ipv6ip

To configure a static IPv6 tunnel interface, use the **tunnel mode ipv6ip** command in interface configuration mode. To remove a static IPv6 tunnel interface, use the **no** form of this command.

tunnel mode ipv6ip [**6rd**| **6to4**| **auto-tunnel**| **isatap**]

no tunnel mode ipv6ip

Syntax Description

6rd	(Optional) Specifies that the tunnel is to be used for IPv6 rapid deployment (6RD).
6to4	(Optional) Configures an IPv6 automatic tunnel using a destination address that is dynamically constructed from an IPv4 address and the prefix 2002::/16 (referred to as a 6to4 address).
auto-tunnel	(Optional) Configures an IPv6 automatic tunnel using an IPv4-compatible IPv6 address.
isatap	(Optional) Configures an IPv6 automatic tunnel using Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) to connect IPv6 nodes (hosts and routers) within IPv4 networks.

Command Default

Static IPv6 tunnel interfaces are not configured.

Command Modes

Interface configuration (config-if)

Command History

Release	Modification
12.2(2)T	This command was introduced.
12.0(21)ST	This command was integrated into Cisco IOS Release 12.0(21)ST.
12.0(22)S	This command was integrated into Cisco IOS Release 12.0(22)S.
12.2(14)S	This command was modified. The **isatap** keyword was added to support the addition of ISATAP tunnel implementation.
12.2(28)SB	This command was integrated into Cisco IOS Release 12.2(28)SB.
12.2(25)SG	This command was integrated into Cisco IOS Release 12.2(25)SG.

Release	Modification
12.2(33)SRA	This command was integrated into Cisco IOS Release 12.2(33)SRA.
12.2(33)SXH	This command was integrated into Cisco IOS Release 12.2(33)SXH.
Cisco IOS XE Release 2.1	This command was integrated into Cisco IOS XE Release 2.1.
Cisco IOS XE Release 3.1S	This command was modified. The **6rd** keyword was added. The **auto-tunnel** keyword was deprecated on Cisco ASR 1000 series routers.
15.1(3)T	This command was integrated into Cisco IOS Release 15.1(3)T.
15.1SY	This command was integrated into Cisco IOS Release 15.1SY. The **auto-tunnel** keyword was deprecated.
Cisco IOS XE Release 3.2SE	This command was integrated into Cisco IOS XE Release 3.2SE.

Usage Guidelines

IPv6 tunneling is the encapsulation of IPv6 packets within IPv4 packets and transmitting the packets across an IPv4 routing infrastructure.

Manually Configured Tunnels

The **tunnel mode ipv6ip** command configures an IPv6 tunnel. The devices at each end of the IPv6 tunnel must support both IPv4 and IPv6 protocol stacks.

To use this command, you must first manually configure the following:

- An IPv6 address on the tunnel interface
- An IPv4 address as the tunnel source
- An IPv4 address as the tunnel destination

Automatic Determination of Tunnel Destination

The **tunnel mode ipv6ip auto-tunnel** command configures an automatic IPv6 tunnel. The tunnel source is manually configured. The tunnel destination is automatically determined as the low-order 32 bits of the IPv4-compatible IPv6 addresses. An IPv4-compatible IPv6 address is a 128-bit IPv6 address that contains the IPv6 prefix 0:0:0:0:0:0 in the high-order 96 bits of the address and an IPv4 address in the low-order 32 bits of the address. The devices at each end of the automatic tunnel must support both IPv4 and IPv6 protocol stacks.

6to4 Tunnels

The **tunnel mode ipv6ip 6to4** command configures an automatic 6to4 tunnel where the tunnel endpoint is determined by a globally unique IPv4 address embedded into a 6to4 address. A 6to4 address is a combination of the prefix 2002::/16 and a globally unique 32-bit IPv4 address. (IPv4-compatible addresses are not used in 6to4 tunneling.) The unique IPv4 address is used as the network-layer address in the 6to4 address prefix. The source of the tunnel is an interface that you can manually configure using the **tunnel source** command. The border devices at each end of a 6to4 tunnel must support both IPv4 and IPv6 protocol stacks. Additionally, the traffic that is destined for the network with the 6to4 address prefix must be routed over the tunnel by using the **ipv6 route** command.

6RD Tunnels

The **tunnel mode ipv6ip 6rd** command specifies that the tunnel is to be used for IPv6 RD. The 6RD feature is similar to the 6to4 tunnel feature, but it does not require addresses to have a 2002::/16 prefix. It also does not require that all 32 bits of the IPv4 destination be in the IPv6 payload header.

ISATAP Tunnels

ISATAP tunnels enable the transportation of IPv6 packets within network boundaries. ISATAP tunnels allow individual IPv4 or IPv6 dual-stack hosts within a site to connect to an IPv6 network using the IPv4 infrastructure.

Unlike IPv4-compatible addresses, ISATAP IPv6 addresses can use any initial unicast /64 prefix. The last 64 bits are used as the interface identifier. Of these, the first 32 bits are the fixed pattern 0000:5EFE. The last 32 bits carry the tunnel endpoint IPv4 address.

Examples

Examples

The following example shows how to configure a manual IPv6 tunnel. In this example, tunnel interface 0 is manually configured with a global IPv6 address. The tunnel source and destination are also manually configured.

```
Device(config)# interface tunnel 0
Device(config-if)# ipv6 address 3ffe:b00:c18:1::3/127
Device(config-if)# tunnel source ethernet 0
Device(config-if)# tunnel destination 192.168.30.1
Device(config-if)# tunnel mode ipv6ip
Device(config-if)# end
```

Examples

The following example shows how to configure an automatic IPv6 tunnel that uses Ethernet interface 0 as the tunnel source. The tunnel destination is determined automatically as the low-order 32 bits of an IPv4-compatible IPv6 address.

```
Device(config)# interface tunnel 0
Device(config-if)# no ip address
Device(config-if)# tunnel source ethernet 0
Device(config-if)# tunnel mode ipv6ip auto-tunnel
Device(config-if)# end
```

Examples

The following example shows how to configure a 6to4 tunnel. In this example, Ethernet interface 0 is configured with an IPv4 address 192.168.99.1. The site-specific 48-bit prefix 2002:c0a8:630 is constructed by prepending the prefix 2002::/16 to the IPv4 address 192.168.99.1.

The tunnel interface 0 is configured without an IPv4 or IPv6 address. The tunnel source address is configured manually as Ethernet interface 0. The tunnel destination address is automatically constructed. An IPv6 static route is configured to route traffic that is destined for network 2002::/16 over tunnel interface 0.

```
Device(config)# interface ethernet 0
Device(config-if)# ip address 192.168.99.1 255.255.255.0
Device(config-if)# ipv6 address 2002:c0a8:6301:1::/64 eui-64
Device(config-if)# exit
Device(config)# interface tunnel 0
Device(config-if)# no ip address
Device(config-if)# ipv6 unnumbered ethernet 0
Device(config-if)# tunnel source ethernet 0
Device(config-if)# tunnel mode ipv6ip 6to4
Device(config-if)# exit
Device(config)# ipv6 route 2002::/16 tunnel 0
Device(config)# end
```

Examples

When a tunnel interface is configured using the **ipv6 unnumbered**, **tunnel source**, and **tunnel mode ipv6ip** commands, the tunnel uses the first IPv6 address configured on the source interface as its IPv6 address. For 6to4 tunnels, the first IPv6 address configured on the source interface must be a 6to4 address. In the following example, the first IPv6 address configured for Ethernet interface 0 (6to4 address 2002:c0a8:6301:1::/64) is used as the IPv6 address of tunnel 0:

```
Device(config)# interface tunnel 0
Device(config-if)# ipv6 unnumbered ethernet 0
Device(config-if)# tunnel source ethernet 0
Device(config-if)# tunnel mode ipv6ip 6to4
Device(config-if)# exit
Device(config)# interface ethernet 0
Device(config-if)# ipv6 address 2002:c0a8:6301:1::/64 eui-64
Device(config-if)# ipv6 address 3ffe:1234:5678::1/64
Device(config-if)# end
```

Examples

The following example shows how to configure a 6RD tunnel:

```
Device(config)# interface Tunnel1
Device(config-if)# ipv6 address 2001:B000:100::1/32
Device(config-if)# tunnel source GigabitEthernet2/0/0
Device(config-if)# tunnel mode ipv6ip 6rd
Device(config-if)# tunnel 6rd prefix 2001:B000::/32
Device(config-if)# tunnel 6rd ipv4 prefix-len 16 suffix-len 8
Device(config-if)# end
Device# show tunnel 6rd Tunnel1

Interface Tunnel1:
  Tunnel Source: 10.1.1.1
  6RD: Operational, V6 Prefix: 2001:B000::/32
       V4 Common Prefix Length: 16, Value: 10.1.0.0
       V4 Common Suffix Length: 8, Value: 0.0.0.1
```

Examples

The following example shows how to configure ISATAP tunnel over an Ethernet interface 0. Router advertisements are enabled to allow client autoconfiguration.

```
Device(config)# interface Ethernet 0
Device(config-if)# ip address 10.1.1.1 255.255.255.0
Device(config)# interface Tunnel 0
Device(config-if)# tunnel source ethernet 0
Device(config-if)# tunnel mode ipv6ip isatap
Device(config-if)# ipv6 address 2001:0DB8::/64 eui-64
Device(config-if)# no ipv6 nd ra suppress
Device(config-if)# end
```

Related Commands

Command	Description
ip address	Specifies the IP address of an IPv4 interface.
ipv6 address	Configures an IPv6 address based on an IPv6 general prefix and enables IPv6 processing on an interface.

Command	Description
ipv6 address eui-64	Configures an IPv6 address for an interface and enables IPv6 processing on the interface using an EUI-64 interface ID in the low-order 64 bits of the address.
ipv6 route	Establishes static IPv6 routes.
ipv6 unnumbered	Enables IPv6 processing on an interface without assigning an explicit IPv6 address to the interface.
no ipv6 nd ra suppress	Reenables the sending of IPv6 router advertisement transmissions on a LAN interface.
show ipv6 interface	Displays the usability status of interfaces configured for IPv6.
show tunnel 6rd tunnel	Displays 6RD information about a tunnel.
tunnel 6rd ipv4	Specifies the prefix length and suffix length of the IPv4 transport address that is common to all the 6RD routers in a domain.
tunnel 6rd prefix	Specifies the common IPv6 prefix on 6RD tunnels.
tunnel destination	Sets the destination address for a tunnel interface.
tunnel source	Sets the source address for a tunnel interface.

vlan configuration

To configure a VLAN or a collection of VLANs and enter VLAN configuration mode, use the **vlan configuration** command in global configuration mode. To return to the command defaults, use the **no** version of this command.

vlan configuration *vlan-id*

Syntax Description

vlan-id	A VLAN or a collection of VLANs.

Command Default

A VLAN or a collection of VLANs is not configured.

Command Modes

Global configuration (config)

Command History

Release	Modification
Cisco IOS XE Release 3.2SE	This command was introduced.

Usage Guidelines

Use the **vlan configuration** command to configure a VLAN or a collection of VLANs. The IPv6 RA throttler, which functions at the VLAN level, counts all RAs from multiple devices over a VLAN during a specified period of time.

Once an IPv6 RA throttler policy has been configured using the **ipv6 nd ra-throttle policy** command, you can attach it to a VLAN or a collection of VLANs using the **ipv6 nd ra-throttle attach-policy** command.

Examples

```
Device(config)# vlan configuration vlan1
Device(config-vlan-config)#
```