



IPv6 Access Control Lists

Access lists determine what traffic is blocked and what traffic is forwarded at device interfaces and allow filtering of traffic based on source and destination addresses, and inbound and outbound traffic to a specific interface. Standard IPv6 ACL functionality was extended to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control. Standard IPv6 ACL functionality was extended to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control.

This module describes how to configure IPv6 traffic filtering and to control access to virtual terminal lines.

- [Prerequisites for IPV6 ACLs, on page 1](#)
- [Restrictions for IPv6 ACLs, on page 1](#)
- [Information About Configuring IPv6 ACLs, on page 2](#)
- [How to Configure IPv6 ACLs, on page 6](#)
- [Configuration Examples for IPv6 ACLs, on page 14](#)
- [Additional References, on page 15](#)
- [Feature Information for IPv6 Access Control Lists, on page 16](#)

Prerequisites for IPV6 ACLs

The following are the prerequisites for IPV6 ACLs:

To use IPv6, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch. You select the template by entering the **sdm prefer {default | dual-ipv4-and-ipv6}** global configuration command.

Restrictions for IPv6 ACLs

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs.

The switch supports most Cisco IOS-supported IPv6 ACLs with some exceptions:

- The switch does not support matching on these keywords: **routing header**, and **undetermined-transport**.
- The switch does not support reflexive ACLs (the **reflect** keyword).
-

-
-
- The switch does not support VLAN ACLs (VLAN maps) for IPv6.
- Output router ACLs and input port ACLs for IPv6 are supported only on switch stacks. Switches support only control plane (incoming) IPv6 ACLs.
- The switch does not apply MAC-based ACLs on IPv6 frames.
- When configuring an ACL, there is no restriction on keywords entered in the ACL, regardless of whether or not they are supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the switch checks to determine whether or not the ACL can be supported on the interface. If not, attaching the ACL is rejected.
- If an ACL is applied to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the switch does not allow the ACE to be added to the ACL that is currently attached to the interface.

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the **fragments** keyword as in IPv4) are supported
- The same statistics supported in IPv4 are supported for IPv6 ACLs.
- If the switch runs out of hardware space, the packets associated with the ACL are dropped on the interface.
- Routed or bridged packets with hop-by-hop options have IPv6 ACLs applied in software.
- Logging is supported for router ACLs, but not for port ACLs.
- The switch supports IPv6 address-matching for a full range of prefix-lengths.

Information About Configuring IPv6 ACLs

You can filter IP version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similarly to the way that you create and apply IP version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic.



Note To use IPv6, you must configure the dual IPv4 and IPv6 Switch Database Management (SDM) template on the switch. You select the template by entering the **sdm prefer {default | dual-ipv4-and-ipv6}** global configuration command.

ACL Overview

Packet filtering can help limit network traffic and restrict network use by certain users or devices. ACLs filter traffic as it passes through a router or switch and permit or deny packets crossing specified interfaces or VLANs. An ACL is a sequential collection of permit and deny conditions that apply to packets. When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. One by one, it tests packets against the conditions in an access list. The first match decides whether the switch

accepts or rejects the packets. Because the switch stops testing after the first match, the order of conditions in the list is critical. If no conditions match, the switch rejects the packet. If there are no restrictions, the switch forwards the packet; otherwise, the switch drops the packet. The switch can use ACLs on all packets it forwards, including packets bridged within a VLAN.

You configure access lists on a router or Layer 3 switch to provide basic security for your network. If you do not configure ACLs, all packets passing through the switch could be allowed onto all parts of the network. You can use ACLs to control which hosts can access different parts of a network or to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow e-mail traffic to be forwarded but not Telnet traffic. ACLs can be configured to block inbound traffic, outbound traffic, or both.

IPv6 ACLs Overview

You can filter IP Version 6 (IPv6) traffic by creating IPv6 access control lists (ACLs) and applying them to interfaces similar to how you create and apply IP Version 4 (IPv4) named ACLs. You can also create and apply input router ACLs to filter Layer 3 management traffic when the switch is running IP base and LAN base feature sets.

A switch supports three types of IPv6 ACLs:

A switch supports two types of IPv6 ACLs:

- IPv6 router ACLs are supported on outbound or inbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. IPv6 router ACLs apply only to IPv6 packets that are routed.
- IPv6 port ACLs are supported on outbound and inbound Layer 2 interfaces. IPv6 port ACLs are applied to all IPv6 packets entering the interface.
- IPv6 router ACLs are supported on outbound or inbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. IPv6 router ACLs apply only to IPv6 packets that are routed.
- IPv6 port ACLs are supported on outbound and inbound Layer 2 interfaces. IPv6 port ACLs are applied to all IPv6 packets entering the interface.
- VLAN ACLs or VLAN maps access-control all packets in a VLAN. You can use VLAN maps to filter traffic between devices in the same VLAN. ACL VLAN maps are applied on L2 VLANs. VLAN maps are configured to provide access control based on Layer 3 addresses for IPv6. Unsupported protocols are access-controlled through MAC addresses using Ethernet ACEs. After a VLAN map is applied to a VLAN, all packets entering the VLAN are checked against the VLAN map.

You can apply both IPv4 and IPv6 ACLs to an interface. As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs.

The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.

Understanding IPv6 ACLs

A switch image supports two types of IPv6 ACLs:

- IPv6 router ACLs - Supported on inbound or outbound traffic on Layer 3 interfaces, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels. Applied to only IPv6 packets that are routed.

- IPv6 port ACLs - Supported on inbound traffic on Layer 2 interfaces only. Applied to all IPv6 packets entering the interface.



Note If you configure unsupported IPv6 ACLs, an error message appears and the configuration does not take affect.

The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.

You can apply both IPv4 and IPv6 ACLs to an interface.

As with IPv4 ACLs, IPv6 port ACLs take precedence over router ACLs:

- When an input router ACL and input port ACL exist in an SVI, packets received on ports to which a port ACL is applied are filtered by the port ACL. Routed IP packets received on other ports are filtered by the router ACL. Other packets are not filtered.
- When an output router ACL and input port ACL exist in an SVI, packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IPv6 packets are filtered by the router ACL. Other packets are not filtered.



Note If any port ACL (IPv4, IPv6, or MAC) is applied to an interface, that port ACL is used to filter packets, and any router ACLs attached to the SVI of the port VLAN are ignored.

Interactions with Other Features and Switches

- If an IPv6 router ACL is configured to deny a packet, the packet is not routed. A copy of the packet is sent to the Internet Control Message Protocol (ICMP) queue to generate an ICMP unreachable message for the frame.
- If a bridged frame is to be dropped due to a port ACL, the frame is not bridged.
- You can create both IPv4 and IPv6 ACLs on a switch or switch stack, and you can apply both IPv4 and IPv6 ACLs to the same interface. Each ACL must have a unique name; an error message appears if you try to use a name that is already configured.

You use different commands to create IPv4 and IPv6 ACLs and to attach IPv4 or IPv6 ACLs to the same Layer 2 or Layer 3 interface. If you use the wrong command to attach an ACL (for example, an IPv4 command to attach an IPv6 ACL), you receive an error message.

- You cannot use MAC ACLs to filter IPv6 frames. MAC ACLs can only filter non-IP frames.
- If the hardware memory is full, packets are dropped on the interface and an unload error message is logged.

Default Configuration for IPv6 ACLs

The default IPv6 ACL configuration is as follows:

```
Switch# show access-lists preauth_ipv6_acl
IPv6 access list preauth_ipv6_acl (per-user)
permit udp any any eq domain sequence 10
```

```
permit tcp any any eq domain sequence 20
permit icmp any any nd-ns sequence 30
permit icmp any any nd-na sequence 40
permit icmp any any router-solicitation sequence 50
permit icmp any any router-advertisement sequence 60
permit icmp any any redirect sequence 70
permit udp any eq 547 any eq 546 sequence 80
permit udp any eq 546 any eq 547 sequence 90
deny ipv6 any any sequence 100
```

Supported ACL Features

IPv6 ACLs on the switch have these characteristics:

- Fragmented frames (the fragments keyword as in IPv4) are supported.
- The same statistics supported in IPv4 are supported for IPv6 ACLs.
- If the switch runs out of TCAM space, packets associated with the ACL label are forwarded to the CPU, and the ACLs are applied in software.
- Routed or bridged packets with hop-by-hop options have IPv6 ACLs applied in software.
- Logging is supported for router ACLs, but not for port ACLs.

IPv6 Port-Based Access Control List Support

The IPv6 PACL feature provides the ability to provide access control (permit or deny) on Layer 2 switch ports for IPv6 traffic. IPv6 PACLs are similar to IPv4 PACLs, which provide access control on Layer 2 switch ports for IPv4 traffic. They are supported only in the ingress direction and in hardware.

A PACL can filter ingress traffic on Layer 2 interfaces based on Layer 3 and Layer 4 header information or non-IP Layer 2 information.

ACLs and Traffic Forwarding

The IPv6 ACL Extensions for Hop by Hop Filtering feature allows you to control IPv6 traffic that might contain hop-by-hop extension headers. You can configure an access control list (ACL) to deny all hop-by-hop traffic or to selectively permit traffic based on protocol.

IPv6 access control lists (ACLs) determine what traffic is blocked and what traffic is forwarded at device interfaces. ACLs allow filtering based on source and destination addresses, inbound and outbound to a specific interface. Use the **ipv6 access-list** command to define an IPv6 ACL, and the **deny** and **permit** commands to configure its conditions.

The IPv6 ACL Extensions for Hop by Hop Filtering feature implements RFC 2460 to support traffic filtering in any upper-layer protocol type.

How to Configure IPv6 ACLs

Configuring IPv6 ACLs

To filter IPv6 traffic, perform this procedure:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **{ipv6 access-list list-name}**
4. **{deny | permit} protocol {source-ipv6-prefix/prefix-length | any} host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any} host destination-ipv6-address} [operator [port-number]][dscp value] [fragments] [log] [log-input] [routing] [sequence value] [time-range name]**
5. **{deny | permit} tcp {source-ipv6-prefix/prefix-length | any} host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any} host destination-ipv6-address} [operator [port-number]] [ack] [dscp value] [established] [fin] [log] [log-input] [neg {port | protocol}] [psh] [range {port | protocol}] [rst] [routing] [sequence value] [syn] [time-range name] [urg]**
6. **{deny | permit} udp {source-ipv6-prefix/prefix-length | any} host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any} host destination-ipv6-address} [operator [port-number]] [dscp value] [log] [log-input] [neg {port | protocol}] [range {port | protocol}] [routing] [sequence value] [time-range name]**
7. **{deny | permit} icmp {source-ipv6-prefix/prefix-length | any} host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any} host destination-ipv6-address} [operator [port-number]] [icmp-type [icmp-code] | icmp-message] [dscp value] [log] [log-input] [routing] [sequence value] [time-range name]**
8. **end**
9. **show ipv6 access-list**
10. **show running-config**
11. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	<p>{ipv6 access-list <i>list-name</i></p> <p>Example:</p> <pre>Switch(config)# ipv6 access-list example_acl_list</pre>	Defines an IPv6 ACL name, and enters IPv6 access list configuration mode.
Step 4	<p>{deny permit} protocol {<i>source-ipv6-prefix/</i> <i>prefix-length</i> any host <i>source-ipv6-address</i>} [operator [<i>port-number</i>]] { <i>destination-ipv6-prefix/</i> <i>prefix-length</i> any host <i>destination-ipv6-address</i>} [operator [<i>port-number</i>]][dscp <i>value</i>] [fragments] [log] [log-input] [routing] [sequence <i>value</i>] [time-range <i>name</i>]</p>	<p>Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions:</p> <ul style="list-style-type: none"> • For protocol, enter the name or number of an IP: ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number. • The <i>source-ipv6-prefix/prefix-length</i> or <i>destination-ipv6-prefix/prefix-length</i> is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons (see RFC 2373). • Enter any as an abbreviation for the IPv6 prefix <i>::/0</i>. • For host <i>source-ipv6-address</i> or <i>destination-ipv6-address</i>, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For operator, specify 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. If the operator follows the <i>source-ipv6-prefix/prefix-length</i> argument, it must match the source port. If the operator follows the <i>destination-ipv6-prefix/prefix-length</i> argument, it must match the destination port. • (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port. You can use TCP port names only when filtering TCP. You can use UDP port names only when filtering UDP. • (Optional) Enter dscp value to match 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. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter routing to specify that IPv6 packets be routed. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4,294,967,295. • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
Step 5	<pre>{deny permit} tcp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6- prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [ack] [dscp value] [established] [fin] [log] [log-input] [neq {port protocol}] [psh] [range {port protocol}] [rst] [routing] [sequence value] [syn] [time-range name] [urg]</pre>	<p>(Optional) Define a TCP access list and the access conditions.</p> <p>Enter tcp for Transmission Control Protocol. The parameters are the same as those described in Step 3a, with these additional optional parameters:</p> <ul style="list-style-type: none"> • ack: Acknowledgment bit set. • established: An established connection. A match occurs if the TCP datagram has the ACK or RST bits set. • fin: Finished bit set; no more data from sender. • neq {port protocol}: Matches only packets that are not on a given port number. • psh—Push function bit set. • range {port protocol}: Matches only packets in the port number range. • rst: Reset bit set. • syn: Synchronize bit set. • urg: Urgent pointer bit set.
Step 6	<pre>{deny permit} udp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [dscp value] [log] [log-input] [neq {port protocol}] [range {port protocol}] [routing] [sequence value] [time-range name]]</pre>	<p>(Optional) Define a UDP access list and the access conditions.</p> <p>Enter udp for the User Datagram Protocol. The UDP parameters are the same as those described for TCP, except that the [operator [port]] port number or name must be a UDP port number or name, and the established parameter is not valid for UDP.</p>

	Command or Action	Purpose
Step 7	<pre>{deny permit} icmp {source-ipv6-prefix/prefix-length any host source-ipv6-address} [operator [port-number]] {destination-ipv6-prefix/prefix-length any host destination-ipv6-address} [operator [port-number]] [icmp-type [icmp-code] icmp-message] [dscp value] [log] [log-input] [routing] [sequence value] [time-range name]</pre>	<p>(Optional) Define an ICMP access list and the access conditions.</p> <p>Enter icmp for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 1, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:</p> <ul style="list-style-type: none"> • <i>icmp-type</i>: Enter to filter by ICMP message type, a number from 0 to 255. • <i>icmp-code</i>: Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. • <i>icmp-message</i>: Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To see a list of ICMP message type names and code names, use the ? key or see command reference for this release.
Step 8	end	Return to privileged EXEC mode.
Step 9	show ipv6 access-list	Verify the access list configuration.
Step 10	show running-config Example: <pre>Switch# show running-config</pre>	Verifies your entries.
Step 11	copy running-config startup-config Example: <pre>Switch# copy running-config startup-config</pre>	(Optional) Saves your entries in the configuration file.

Attaching an IPv6 ACL to an Interface

You can apply an ACL to outbound or inbound traffic on Layer 3 interfaces, or to inbound traffic on Layer 2 interfaces. You can also apply ACLs only to inbound management traffic on Layer 3 interfaces.

Follow these steps to control access to an interface:

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-id*

4. **no switchport**
5. **ipv6 address** *ipv6-address*
6. **ipv6 traffic-filter** *access-list-name* {**in** | **out**}
7. **end**
8. **show running-config**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Switch> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Switch# configure terminal	Enters global configuration mode.
Step 3	interface <i>interface-id</i>	Identify a Layer 2 interface (for port ACLs) or Layer 3 interface (for router ACLs) on which to apply an access list, and enter interface configuration mode.
Step 4	no switchport	If applying a router ACL, this changes the interface from Layer 2 mode (the default) to Layer 3 mode.
Step 5	ipv6 address <i>ipv6-address</i>	Configure an IPv6 address on a Layer 3 interface (for router ACLs).
Step 6	ipv6 traffic-filter <i>access-list-name</i> { in out }	Apply the access list to incoming or outgoing traffic on the interface. Note The out keyword is not supported for Layer 2 interfaces (port ACLs).
Step 7	end Example: Switch(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Switch# show running-config	Verifies your entries.

	Command or Action	Purpose
Step 9	copy running-config startup-config Example: Switch# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Monitoring IPv6 ACLs

You can display information about all configured access lists, all IPv6 access lists, or a specific access list by using one or more of the privileged EXEC commands shown in the table below:

Command	Purpose
show access-lists	Displays all access lists configured on the switch.
show ipv6 access-list [<i>access-list-name</i>]	Displays all configured IPv6 access lists or the access list specified by name.

This is an example of the output from the **show access-lists** privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack.

```
Switch # show access-lists
Extended IP access list hello
 10 permit ip any any
IPv6 access list ipv6
  permit ipv6 any any sequence 10
```

This is an example of the output from the **show ipv6 access-list** privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack

```
Switch# show ipv6 access-list
IPv6 access list inbound
  permit tcp any any eq bgp (8 matches) sequence 10
  permit tcp any any eq telnet (15 matches) sequence 20
  permit udp any any sequence 30

IPv6 access list outbound
  deny udp any any sequence 10
  deny tcp any any eq telnet sequence 20
```

Configuring PACL Mode and Applying IPv6 PACL on an Interface

Before you begin

Before you configure the IPv6 PACL feature, you must configure an IPv6 access list. Once you have configured the IPv6 access list, you must configure the port-based access control list (PACL) mode on the specified IPv6 Layer 2 interface.

SUMMARY STEPS

1. **enable**

2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. **exit**
5. **interface** *type number*
6. **ipv6 traffic-filter** *access-list-name* {**in** | **out**}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 access-list <i>access-list-name</i> Example: Device(config)# ipv6 access-list list1	Defines an IPv6 ACL and enters IPv6 access list configuration mode.
Step 4	exit Example: Device(config-ipv6-acl)# exit	Exits IPv6 access list configuration mode and enters global configuration mode.
Step 5	interface <i>type number</i> Example:	Specifies an interface type and number and enters interface configuration mode.
Step 6	ipv6 traffic-filter <i>access-list-name</i> { in out }	Filters incoming and outgoing IPv6 traffic on an interface.
Step 7	end Example: Device(config-if)# end	Exits interface configuration mode and enters privileged EXEC mode.

Configuring IPv6 ACL Extensions for Hop by Hop Filtering

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*

4. **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** [*header-number* | *header-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*]
5. **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** [*header-number* | *header-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**]
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 access-list <i>access-list-name</i> Example: Device(config)# ipv6 access-list hbh-acl	Defines an IPv6 ACL and enters IPv6 access list configuration mode.
Step 4	permit <i>protocol</i> { <i>source-ipv6-prefix/prefix-length</i> any host <i>source-ipv6-address</i> auth } [<i>operator</i> [<i>port-number</i>]] { <i>destination-ipv6-prefix/prefix-length</i> any host <i>destination-ipv6-address</i> auth } [<i>operator</i> [<i>port-number</i>]] [dest-option-type [<i>header-number</i> <i>header-type</i>]] [dscp <i>value</i>] [flow-label <i>value</i>] [fragments] [hbh] [log] [log-input] [mobility] [mobility-type [<i>mh-number</i> <i>mh-type</i>]] [reflect <i>name</i>] [timeout <i>value</i>] [routing] [routing-type <i>routing-number</i>] [sequence <i>value</i>] [time-range <i>name</i>] Example: Device(config-ipv6-acl)# permit icmp any any dest-option-type	Sets permit conditions for the IPv6 ACL.
Step 5	deny <i>protocol</i> { <i>source-ipv6-prefix/prefix-length</i> any host <i>source-ipv6-address</i> / auth } [<i>operator</i> [<i>port-number</i>]] { <i>destination-ipv6-prefix/prefix-length</i> any host <i>destination-ipv6-address</i> auth } [<i>operator</i> [<i>port-number</i>]] [dest-option-type [<i>header-number</i> <i>header-type</i>]] [dscp <i>value</i>] [flow-label <i>value</i>] [fragments] [hbh] [log] [log-input] [mobility] [mobility-type [<i>mh-number</i> <i>mh-type</i>]] [routing] [routing-type <i>routing-number</i>]	Sets deny conditions for the IPv6 ACL.

	Command or Action	Purpose
	<p>[sequence <i>value</i>] [time-range <i>name</i>] [undetermined-transport]</p> <p>Example:</p> <pre>Device(config-ipv6-acl)# deny icmp any any dest-option-type</pre>	
Step 6	<p>end</p> <p>Example:</p> <pre>Device (config-ipv6-acl)# end</pre>	Returns to privileged EXEC configuration mode.

Configuration Examples for IPv6 ACLs

Example: Configuring IPv6 ACLs

This example configures the IPv6 access list named CISCO. The first deny entry in the list denies all packets that have a destination TCP port number greater than 5000. The second deny entry denies packets that have a source UDP port number less than 5000. The second deny also logs all matches to the console. The first permit entry in the list permits all ICMP packets. The second permit entry in the list permits all other traffic. The second permit entry is necessary because an implicit deny -all condition is at the end of each IPv6 access list.

```
Switch(config)# ipv6 access-list CISCO
Switch(config-ipv6-acl)# deny tcp any any gt 5000
Switch config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Switch(config-ipv6-acl)# permit icmp any any
Switch(config-ipv6-acl)# permit any any
```

Example: Applying IPv6 ACLs

This example shows how to apply the access list Cisco to outbound traffic on a Layer 3 interface.

```
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```

Example: Configuring PACL Mode and Applying IPv6 PACL on an Interface

```
Device# configure terminal
Device(config)# ipv6 access-list list1
Device(config-ipv6-acl)# exit
Device(config-if)# ipv6 traffic-filter list1 in
```

Example: IPv6 ACL Extensions for Hop by Hop Filtering

```

Device(config)# ipv6 access-list hbh_acl
Device(config-ipv6-acl)# permit tcp any any hbh
Device(config-ipv6-acl)# permit tcp any any
Device(config-ipv6-acl)# permit udp any any
Device(config-ipv6-acl)# permit udp any any hbh
Device(config-ipv6-acl)# permit hbh any any
Device(config-ipv6-acl)# permit any any
Device(config-ipv6-acl)# hardware statistics
Device(config-ipv6-acl)# exit

! Assign an IP address and add the ACL on the interface.

Device(config)# interface FastEthernet3/1
Device(config-if)# ipv6 address 1001::1/64
Device(config-if)# ipv6 traffic-filter hbh_acl in
Device(config-if)# exit
Device(config)# exit
Device# clear counters
Clear "show interface" counters on all interfaces [confirm]
Device#

! Verify the configurations.

Device# show running-config interface FastEthernet3/1

Building configuration...

Current configuration : 114 bytes
!
interface FastEthernet3/1
no switchport
ipv6 address 1001::1/64
ipv6 traffic-filter hbh_acl
end

```

Additional References

Related Documents

MIBs

MIB	MIBs Link
All the supported MIBs for this release.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
<p>The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies.</p> <p>To receive security and technical information about your products, you can subscribe to various services, such as the Product Alert Tool (accessed from Field Notices), the Cisco Technical Services Newsletter, and Really Simple Syndication (RSS) Feeds.</p> <p>Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.</p>	http://www.cisco.com/support

Feature Information for IPv6 Access Control Lists

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1: Feature Information for IPv6 Access Control Lists

Feature Name	Releases	Feature Information
IPv6 ACL Extensions for Hop-by-Hop Filtering	15.1(1)SG	<p>Allows you to control IPv6 traffic that might contain hop-by-hop extension headers.</p> <p>This feature was supported on CAT3560C, CAT3560CX, CAT3560X, CAT3750X, CAT4500-X.</p> <p>The following commands were introduced or modified: deny (IPv6), permit (IPv6).</p>

Feature Name	Releases	Feature Information
IPv6 PACL Support		<p>The IPv6 PACL feature permits or denies the movement of traffic between port-based interface, Layer 3 subnets, wireless or wired clients, and VLANs, or within a VLAN.</p> <p>This feature was supported on CAT2960, CAT2960S, CAT3560X, CAT3650, CAT3560CX, CAT4500.</p> <p>The following command was introduced or modified: ipv6 traffic-filter.</p>
IPv6 Services: Extended Access Control Lists	12.2(25)SG	Standard IPv6 ACL functionality was extended to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control.
IPv6 Services: Standard Access Control Lists	12.2(25)SG	Access lists determine what traffic is blocked and what traffic is forwarded at router interfaces and allow filtering based on source and destination addresses, inbound and outbound to a specific interface.

