Configuring IPv6 ACLs

This chapter provides details about configuring IPv6 access control lists (ACLs) on the Cisco Industrial Ethernet Switches, hereafter referred to as *switch*.

When the switch is running the IP services image:

- You can filter IPv6 traffic by creating IPv6 ACLs and applying them to interfaces
- You can create and apply input router ACLs to filter Layer 3 management traffic

This chapter contains the following sections:

- Information About IPv6 ACLs, page 829
- Prerequisites, page 830
- Guidelines and Limitations, page 830
- Default Settings, page 831
- Configuring IPv6 ACLs, page 831
- Verifying IPv6 ACLs, page 835
- Configuration Example, page 836

Information About IPv6 ACLs

A switch running the IP services image supports two types of IPv6 ACLs:

- IPv6 *router ACLs* on outbound or inbound traffic on Layer 3 interfaces only, which can be routed ports, switch virtual interfaces (SVIs), or Layer 3 EtherChannels.

  IPv6 router ACLs apply only to routed IPv6 packets.

- *IPv6 port ACLs* on inbound traffic on Layer 2 interfaces only. The switch applies IPv6 port ACLs to all IPv6 packets entering the interface.

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: When you apply *any* port ACL (IPv4, IPv6, or MAC) to an interface, that port ACL filters packets, and ignores any router ACLs attached to the SVI of the port VLAN.

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 hardware space, packets associated with the ACL are forwarded to the CPU, and the software applies the ACLs.
- 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.

Note: For items not supported for IPv6 ACLS, see Guidelines and Limitations, page 830.

Prerequisites

Be sure to review Guidelines and Limitations, page 830 and the Before You Begin section within each configuration section before configuring a feature.

Guidelines and Limitations

ACLs for IPv6 Traffic Not Supported

- The switch does not support VLAN ACLs (VLAN maps) for IPv6 traffic.
- The switch does not apply MAC-based ACLs on IPv6 frames.
- You cannot apply IPv6 port ACLs to Layer 2 EtherChannels.
- The switch does not support output port ACLs.

Cisco IOS IPv6 ACLs Functions Not Supported

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

Access Control Entry (ACE) and ACLs

- When you apply an ACL 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 attached ACL.

Named ACLs

- IPv6 supports only named ACLs.

IPv6 ACLs Interactions With Other Switches or Features

- When you configure an IPv6 router ACL to deny a packet, the software does not route the packet. Instead, the software forwards a copy of the packet 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, and you can apply both IPv4 and IPv6 ACLs to the same interface.
  - Each ACL must have a unique name; and, an error message appears if you try to use a name that already exists on the switch.
  - 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, for any additional configured ACLs, the switch forwards the packets to the CPU, and the software applies the ACLs.

Default Settings

Parameters	Default
IPv6 ACLs	There are no default IPv6 ACLs configured or applied on the switch.

Configuring IPv6 ACLs

This section includes the following topics:

- Creating IPv6 ACLs, page 831
- Applying an IPv6 ACL to an Interface, page 835

BEFORE YOU BEGIN

Review the Guidelines and Limitations, page 830 for this feature.

Select one of the dual IPv4 and IPv6 SDM templates.

Creating IPv6 ACLs

Note: When you configure an unsupported IPv6 ACL, an error message appears, and the configuration does not take affect.

Use the **no** {**deny** | **permit**} IPv6 access-list configuration commands with keywords to remove the deny or permit conditions from the specified access list for the commands below.

Configuring IPv6 ACLs

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**ipv6 access-list** *access-list-name*	Define an IPv6 access list using a name, and enter IPv6 access-list configuration mode.
3. a	{**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*]	Deny or permit the packet, when specified conditions are matched. These are the conditions: ■ *protocol*–Name or number of an Internet protocol: **ahp**, **esp**, **icmp**, **ipv6**, **pcp**, **stcp**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IPv6 protocol number. For additional specific parameters for ICMP, TCP, and UDP, see Steps 3b through 3d. ■ *source-ipv6-prefix*/*prefix-length* or *destination-ipv6-prefix*/ *prefix-length*–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. ■ Enter **any** as an abbreviation for the IPv6 prefix ::/0. ■ **host** *source-ipv6-address* or *destination-ipv6-address*– Define source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons. ■ (Optional) *operator*–Operand that compares the source or destination ports of the specified protocol such as **lt** (less than), **gt** (greater than), **eq** (equal), **neq** (not equal), and **range**. If the operator follows the *source-ipv6-prefix*/*prefix-length* argument, it must match the source port. If the operator follows the *destination-ipv6- prefix*/*prefix-length* argument, it must match the destination port.

Configuring IPv6 ACLs

	Command	Purpose
		■ (Optional) *port-number*– Value of 0 to 65535 or TCP or UDP port name. Use TCP port names only when filtering TCP. Use UDP port names only when filtering UDP. ■ (Optional) **dscp** *value*–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) **fragments**–Check noninitial fragments. Keyword is only visible when the protocol is **ipv6**. ■ (Optional, router ACLs only) **log**–Send a logging message to the console about the packet that matches the entry. Enter **log-input** to include the input interface in the log entry. ■ (Optional) **routing**–Specify routing of IPv6 packets. ■ (Optional) **sequence** *value*–Specify the sequence number for the access list statement. Value range is from 1 to 4294967295. ■ (Optional) **time-range** *name*–Specify the time range that applies to the deny or permit statement.
Step 3b	{**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**]	(Optional) Define a TCP access list and the access conditions. Enter **tcp** for Transmission Control Protocol. The parameters are the same as those described in Step 3a, with these additional optional parameters: ■ **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*}–Match only packets that are not on a given port number. ■ **psh**–Push function bit set. ■ **range** {*port* | *protocol*}–Match only packets in the port number range. ■ **rst**–Reset bit set. ■ **syn**–Synchronize bit set. ■ **urg**–Urgent pointer bit set.

Configuring IPv6 ACLs

	Command	Purpose
Step 3c	{**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*]	(Optional) Define a UDP access list and the access conditions. 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.
Step 3d	{**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*]	(Optional) Define an ICMP access list and the access conditions. Enter **icmp** for Internet Control Message Protocol. The ICMP parameters are the same as those described for most IP protocols in Step 3a, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings: ■ *icmp-type*—Enter to filter by ICMP message type, a number from 0 to 255. ■ *icmp-code*—Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255. ■ *icmp-message*—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.
4.	**end**	Return to privileged EXEC mode.
5.	**show ipv6 access-list**	Verify the access list configuration.
6.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

The following example:

- Creates an IPv6 ACL named CISCO.
- Defines one deny entry that denies all packets that have a destination TCP port number greater than 5000 and a second deny entry that denies packets that have a source UDP port number less than 5000. The second deny entry also logs all matches to the console.
- Defines a permit entry to permit all ICMP packets and another permit entry that allows 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
```

Applying an IPv6 ACL to an Interface

BEFORE YOU BEGIN

Review the Guidelines and Limitations, page 830 for this feature.

DETAILED STEPS

	Command	Purpose
1.	**configure terminal**	Enter global configuration mode.
2.	**interface** *interface-id*	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.
3.	**no switchport**	If applying a router ACL, change the interface from Layer 2 mode (the default) to Layer 3 mode.
4.	**ipv6 address** *ipv6-address*	Configure an IPv6 address on a Layer 3 interface (for router ACLs). **Note:** This command is not required on Layer 2 interfaces or if the interface is already configured with an explicit IPv6 address. Use the **no ipv6 traffic-filter** *access-list-name* interface configuration command to remove an access list from an interface.
5.	**ipv6 traffic-filter** *access-list-name* {**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).
6.	**end**	Return to privileged EXEC mode.
7.	**show running-config**	Verify the access list configuration.
8.	**copy running-config startup-config**	(Optional) Save your entries in the configuration file.

EXAMPLE

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

```
Switch(config)# interface gigabitethernet 0/3
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```

Verifying 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 following privileged EXEC commands.

Command	Purpose
show access-lists	Display all access lists configured on the switch.
show ipv6 access-list [*access-list-name*]	Display all configured IPv6 access list or the access list specified by name.

Configuration Example

The following example:

- Creates an IPv6 ACL named CISCO.
- Defines one deny entry that denies all packets that have a destination TCP port number greater than 5000 and a second deny entry that denies packets that have a source UDP port number less than 5000. The second deny entry also logs all matches to the console.
- Defines a permit entry to permit all ICMP packets and another permit entry that allows all other traffic. The second permit entry is necessary because an implicit deny-all condition is at the end of each IPv6 access list.
- Applies the access list CISCO to outbound traffic on a Layer 3 interface.

```
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
Switch(config-ipv6-acl)# exit
Switch(config)# interface gigabitethernet 0/3
Switch(config-if)# no switchport
Switch(config-if)# ipv6 address 2001::/64 eui-64
Switch(config-if)# ipv6 traffic-filter CISCO out
```