



IPv6 ACLs

- [Restrictions for IPv6 ACLs, on page 1](#)
- [IPv6 ACLs Overview, on page 2](#)
- [Default Configuration for IPv6 ACLs , on page 6](#)
- [Configuring IPv6 ACLs, on page 6](#)
- [Attaching an IPv6 ACL to an Interface, on page 10](#)
- [Configuring a VLAN Map, on page 11](#)
- [Applying a VLAN Map to a VLAN, on page 13](#)
- [Monitoring IPv6 ACLs, on page 14](#)
- [Configuration Examples for IPv6 ACL, on page 15](#)
- [Additional References, on page 18](#)
- [Feature Information for IPv6 ACLs, on page 18](#)

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).
- This release supports port ACLs, router ACLs and VLAN ACLs (VLAN maps) for IPv6.
- 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.
- Logging is supported for router ACLs, but not for port ACLs.
- The switch supports IPv6 address-matching for a full range of prefix-lengths.
- If a downloadable ACL contains any type of duplicate entries, the entries are not auto merged. As a result, the 802.1X session authorization fails. Ensure that the downloadable ACL is optimized without any duplicate entries, for example port-based and name-based entries for the same port.

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:

- 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.

The switch supports 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.

Understanding 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 inbound traffic on Layer 2 interfaces only. IPv6 port ACLs are applied to all IPv6 packets entering the interface.

A switch running the IP base feature set supports only input router IPv6 ACLs. It does not support port ACLs or output IPv6 router ACLs.



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

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.

Types of ACL

Per User IPv6 ACL

For the per-user ACL, the full access control entries (ACE) as the text strings are configured on the Cisco Secure Access Control Server (Cisco Secure ACS).

Filter ID IPv6 ACL

For the filter-Id ACL, the full ACEs and the `acl_name(filter-id)` is configured on the device and only the `filter-id` is configured on the Cisco Secure ACS.

Downloadable IPv6 ACL

For the downloadable ACL (dACL), all the full ACEs and the `dac1` name are configured only on the Cisco Secure ACS.

The Cisco Secure ACS sends the `dac1` name to the device in its `ACCESS-Accept` attribute, which takes the `dac1` name and sends the `dACL` name back to the Cisco Secure ACS for the ACEs, using the `ACCESS-request` attribute.

Switch Stacks and IPv6 ACLs

The active switch supports IPv6 ACLs in hardware and distributes the IPv6 ACLs to the stack members.

If a standby switch takes over as the active switch, it distributes the ACL configuration to all stack members. The member switches sync up the configuration distributed by the new active switch and flush out entries that are not required.

When an ACL is modified, attached to, or detached from an interface, the active switch distributes the change to all stack members.

ACL Precedence

When VLAN maps, Port ACLs, and router ACLs are configured on the same switch, the filtering precedence, from greatest to least for ingress traffic is port ACL, VLAN map, and then router ACL. For egress traffic, the filtering precedence is router ACL, VLAN map, and then port ACL.

The following examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL applied are filtered by the port ACL. Other packets are filtered by the VLAN map
- When an input router ACL and input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports to which a port ACL is applied are filtered by the port ACL. Incoming 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, incoming packets received on the ports to which a port ACL is applied are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, input router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, output router ACL, and input port ACL exist in an SVI, incoming packets received on the ports to which a port ACL is applied are only filtered by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

VLAN Maps

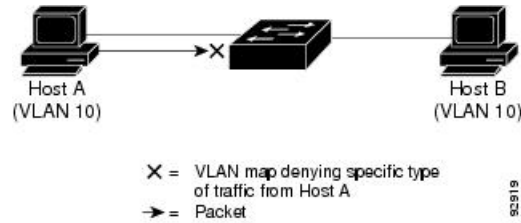
VLAN ACLs or VLAN maps are used to control network traffic within a VLAN. You can apply VLAN maps to all packets that are bridged within a VLAN in the switch or switch stack. VACLs are strictly for security packet filtering and for redirecting traffic to specific physical interfaces. VACLs are not defined by direction (ingress or egress).

All non-IP protocols are access-controlled through MAC addresses and Ethertype using MAC VLAN maps. (IP traffic is not access controlled by MAC VLAN maps.) You can enforce VLAN maps only on packets going through the switch; you cannot enforce VLAN maps on traffic between hosts on a hub or on another switch connected to this switch.

With VLAN maps, forwarding of packets is permitted or denied, based on the action specified in the map.

Figure 1: Using VLAN Maps to Control Traffic

This figure shows how a VLAN map is applied to prevent a specific type of traffic from Host A in VLAN 10 from being forwarded. You can apply only one VLAN map to a VLAN.



Hitless TCAM Update

The Hitless TCAM update for IPv4 and IPv6 provides the capability to apply existing features to the incoming traffic while updating new features in the TCAM. Any change in IPv4 and IPv6 ACL on a given interface would trigger a reprogramming of TCAM.

Starting with Cisco IOS XE Fuji 16.8.1a, Hitless TCAM update is enabled.

This feature is always enabled. You cannot disable this feature.

The Hitless TCAM update follows the below ACL change rules:

- If there are value compare unit (VCU) registers in use from ACEs with layer 4 operators, there could be a few packet drops during the change.
- If there are not enough VCU bits remaining to add a second set of access control entries and if there is not enough space in TCAM to expand these entries, the old ACL change method will apply; which will drop all packets, delete the old ACL, add the new ACL entries into TCAM, and then remove the entry that is causing the packets to drop.
- If there is not enough space in TCAM to add the modified entries, the old ACL change method will automatically be applied.



Note

- To perform Hitless ACL update for an IPv4 ACL which has X number of ACEs, TCAM should have a free space for accommodating X+1 entries.
- To perform Hitless ACL update for an IPv6 ACL which has X number of ACEs, TCAM should have a free space for accommodating 2X+2 entries.

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
```

Configuring IPv6 ACLs

To filter IPv6 traffic, perform this procedure:

Procedure

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

	Command or Action	Purpose
Step 3	<p>[no]{ipv6 access-list <i>list-name</i> client permit-control-packets log-update threshold role-based <i>list-name</i>}</p> <p>Example:</p> <pre>Device(config)# ipv6 access-list example_acl_list</pre>	<p>Defines an IPv6 ACL name, and enters IPv6 access list configuration mode.</p>
Step 4	<p>[no]{deny permit} protocol {<i>source-ipv6-prefix/ prefix-length</i> any threshold host <i>source-ipv6-address</i>} [operator [<i>port-number</i>]] { <i>destination-ipv6-prefix/ 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 ::/0. • 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. <p>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.</p> <ul style="list-style-type: none"> • (Optional) The port-number is a decimal number from 0 to 65535 or the name of

	Command or Action	Purpose
		<p>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.</p> <ul style="list-style-type: none"> • (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. • (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.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • neq { <i>port</i> protocol }: Matches only packets that are not on a given port number. • psh—Push function bit set. • range { <i>port</i> 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>
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.

	Command or Action	Purpose
Step 10	show running-config Example: Device# <code>show running-config</code>	Verifies your entries.
Step 11	copy running-config startup-config Example: Device# <code>copy running-config startup-config</code>	(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.

Procedure

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

	Command or Action	Purpose
Step 6	ipv6 traffic-filter <i>access-list-name</i> {in out}	Apply the access list to incoming or outgoing traffic on the interface.
Step 7	end Example: Device(config)# end	Returns to privileged EXEC mode.
Step 8	show running-config Example: Device# show running-config	Verifies your entries.
Step 9	copy running-config startup-config Example: Device# copy running-config startup-config	(Optional) Saves your entries in the configuration file.

Configuring a VLAN Map

To create a VLAN map and apply it to one or more VLANs, perform these steps:

Before you begin

Create the IPv6 ACL that you want to apply to the VLAN.

Procedure

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

	Command or Action	Purpose
Step 3	<p>vlan access-map <i>name</i> [<i>number</i>]</p> <p>Example:</p> <pre>Device(config)# vlan access-map map_1 20</pre>	<p>Creates a VLAN map, and give it a name and (optionally) a number. The number is the sequence number of the entry within the map.</p> <p>When you create VLAN maps with the same name, numbers are assigned sequentially in increments of 10. When modifying or deleting maps, you can enter the number of the map entry that you want to modify or delete.</p> <p>VLAN maps do not use the specific permit or deny keywords. To deny a packet by using VLAN maps, create an ACL that would match the packet, and set the action to drop. A permit in the ACL counts as a match. A deny in the ACL means no match.</p> <p>Entering this command changes to access-map configuration mode.</p>
Step 4	<p>match {<i>ip</i> <i>ipv6</i> <i>mac</i>} address {<i>name</i> <i>number</i>} [<i>name</i> <i>number</i>]</p> <p>Example:</p> <pre>Device(config-access-map)# match ipv6 address ip_net</pre>	<p>Match the packet against one or more access lists. Note that packets are only matched against access lists of the correct protocol type. IP packets are matched against IP access lists. Non-IP packets are only matched against named MAC access lists.</p> <p>Note If the VLAN map is configured with a match clause for a type of packet (IP or MAC) and the map action is drop, all packets that match the type are dropped. If the VLAN map has no match clause, and the configured action is drop, all IP and Layer 2 packets are dropped.</p>
Step 5	<p>Enter one of the following commands to specify an IP packet or a non-IP packet (with only a known MAC address) and to match the packet against one or more ACLs:</p> <ul style="list-style-type: none"> • action { forward } <pre>Device(config-access-map)# action forward</pre> <ul style="list-style-type: none"> • action { drop } <pre>Device(config-access-map)# action drop</pre>	<p>Sets the action for the map entry.</p>

	Command or Action	Purpose
Step 6	vlan filter <i>mapname</i> vlan-list <i>list</i> Example: <pre>Device(config)# vlan filter map 1 vlan-list 20-22</pre>	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.

Applying a VLAN Map to a VLAN

To apply a VLAN map to one or more VLANs, perform these steps.

Procedure

	Command or Action	Purpose
Step 1		
Step 2	configure terminal Example: <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	vlan filter <i>mapname</i> vlan-list <i>list</i> Example: <pre>Device(config)# vlan filter map 1 vlan-list 20-22</pre>	Applies the VLAN map to one or more VLAN IDs. The list can be a single VLAN ID (22), a consecutive list (10-22), or a string of VLAN IDs (12, 22, 30). Spaces around the comma and hyphen are optional.
Step 4	end Example: <pre>Device(config)# end</pre>	Returns to privileged EXEC mode.
Step 5	show running-config Example: <pre>Device# show running-config</pre>	Displays the access list configuration.
Step 6	copy running-config startup-config Example: <pre>Device# copy running-config</pre>	(Optional) Saves your entries in the configuration file.

	Command or Action	Purpose
	startup-config	

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:

Table 1: show ACL commands

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.
show vlan access-map [<i>map-name</i>]	Displays VLAN access map configuration.
show vlan filter [access-map <i>access-map</i> vlan <i>vlan-id</i>]	Displays the mapping between VACLs and VLANs.

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
```

This is an example of the output from the `show vlan access-map` privileged EXEC command. The output shows VLAN access map information.

```
Switch# show vlan access-map
Vlan access-map "m1" 10
Match clauses:
  ipv6 address: ip2
Action: drop
```

Configuration Examples for IPv6 ACL

Example: Creating an IPv6 ACL

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.



Note Logging is supported only on Layer 3 interfaces.

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

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

Example: Displaying IPv6 ACLs

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.

```
Device #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-lists** privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack.

```
Device# 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 RA Guard Policy

Procedure

	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 nd raguard policy <i>policy name</i> Example: Device (config)# ipv6 nd raguard policy MyPolicy	
Step 4	trusted-port Example: Device (config-nd-raguard)# trusted-port	Configures the trusted port for the policy created above.
Step 5	device-role router Example: Device (config-nd-raguard)# device-role [host monitor router switch] Device (config-nd-raguard)# device-role router d	Defines the trusted device that can send RAs to the trusted port created above.
Step 6	interface <i>interface-id</i> Example: Device (config)# interface tenGigabitEthernet 1/0/1	Configures the interface to the trusted device.
Step 7	ipv6 nd raguard attach-policy <i>policy name</i> Example: Device (config-if)# ipv6 nd raguard attach-policy Mypolicy	Configures and attaches the policy to trust the RA's received from the port.
Step 8	vlan <i>vlan-id</i> Example: Device (config)# vlan configuration 19-21,23	Configures the wireless client vlans.
Step 9	ipv6 nd suppress Example:	Suppresses the ND messages over wireless.

	Command or Action	Purpose
	Device(config-vlan-config)# ipv6 nd suppress	
Step 10	ipv6 snooping Example: Device(config-vlan-config)# ipv6 snooping	Captures IPv6 traffic.
Step 11	ipv6 nd rguard attach-policy <i>policy name</i> Example: Device(config-vlan-config)# ipv6 nd rguard attach-policy Mypolicy	Attaches the RA Guard policy to the wireless client vlans.
Step 12	ipv6 nd ra-throttler attach-policy <i>policy name</i> Example: Device(config-vlan-config)# ipv6 nd ra-throttler attach-policy Mythrottle	Attaches the RA throttling policy to the wireless client vlans.

Configuring IPv6 Neighbor Binding

Procedure

	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 neighbor binding [vlan] 19 2001:db8::25:4 interface tenGigabitEthernet 1/0/3 aaa.bbb.ccc Example: Device(config)# ipv6 neighbor binding vlan 19 2001:db8::25:4 interface tenGigabitEthernet 1/0/3 aaa.bbb.ccc	Sets and validates the neighbor 2001:db8::25:4 only valid when transmitting on VLAN 19 through interface te1/0/3 with the source mac-address as aaa.bbb.ccc.

Additional References

Related Documents

Error Message Decoder

Description	Link
To help you research and resolve system error messages in this release, use the Error Message Decoder tool.	https://www.cisco.com/cgi-bin/Support/Errordecoder/index.cgi

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 ACLs

This table lists the features in this module and provides links to specific configuration information:

Feature	Release	Modification
IPv6 ACL Functionality	Cisco IOS XE 3.3SE Cisco IOS XE 3.3SE	This feature was introduced.
Downloadable IPv6 ACL	Cisco IOS XE Gibraltar 16.11.1	This feature was introduced.