Configuring IPv6 ACL

Prerequisites for IPv6 ACL

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 when the switch is running the IP base feature set.

Related Topics

Restrictions for IPv6 ACL

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 of the Cisco IOS-supported IPv6 ACLs with some exceptions:

- 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).

- 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

Information About IPv6 ACL

An access control list (ACL) is a set of rules used to limit access to a particular interface (for example, if you want to restrict a wireless client from pinging the management interface of the controller). ACLs are configured on the switchnd applied to the management interface, the AP-manager interface, any of the dynamic interfaces, or a WLAN to control data traffic to and from wireless clients or to the controller central processing unit (CPU) to control all traffic destined for the CPU.

You can also create a preauthentication ACL for web authentication. Such an ACL is used to allow certain types of traffic before authentication is complete.

IPv6 ACLs support the same options as IPv4 ACLs including source, destination, source and destination ports.

Note You can enable only IPv4 traffic in your network by blocking IPv6 traffic. That is, you can configure an IPv6 ACL to deny all IPv6 traffic and apply it on specific or all WLANs.

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 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.

Related Topics

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 ACS.

The ACE is not configured on the Controller. The ACE is sent to the switch in the `ACCESS-Accept` attribute and applies it directly for the client. When a wireless client roams into an foreign switch, the ACEs are sent to the foreign switch as an AAA attribute in the mobility Handoff message. Output direction, using per-user ACL is not supported.

Filter ID IPv6 ACL

For the filter-Id ACL, the full ACEs and the `acl name(filter-id)` is configured on the switch and only the `filter-id` is configured on the ACS. The `filter-id` is sent to the switch in the ACCESS-Accept attribute, and the switch looks up the filter-id for the ACEs, and then applies the ACEs to the client. When the client L2 roams to the foreign switch, only the filter-id is sent to the foreign switch in the mobility Handoff message. Output filtered ACL, using per-user ACL is not supported. The foreign switch has to configure the filter-id and ACEs beforehand.

Downloadable IPv6 ACL

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

> **Note** The controller does not configure any ACL.

The ACS sends the `dacl` name to the switch in its `ACCESS-Accept` attribute, which takes the `dacl` name and sends the `dACL` name back to the ACS, for the ACEs, using the `access-request` attribute.

The ACS responds to the corresponding ACEs of the switch in the `access-accept` attribute. When the wireless client roams to an foreign switch, only the `dacl` name is sent to the foreign switch in the mobility Handoff message. The foreign switch contacts the ACS server with the `dacl` name to retrieve the ACEs.

IPv6 ACLs and Switch Stacks

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

> **Note** For full IPv6 functionality in a switch stack, all stack members must be running the IP services feature set.

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

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

Configuring IPv6 ACLs

To filter IPv6 traffic, you perform these steps:

Before You Begin

Before configuring IPv6 ACLs, you must select one of the dual IPv4 and IPv6 SDM templates.

SUMMARY STEPS

1. Create an IPv6 ACL, and enter IPv6 access list configuration mode.
2. Configure the IPv6 ACL to block (deny) or pass (permit) traffic.
3. Apply the IPv6 ACL to the interface where the traffic needs to be filtered.
4. Apply the IPv6 ACL to an interface. For router ACLs, you must also configure an IPv6 address on the Layer 3 interface to which the ACL is applied.

DETAILED STEPS

	Command or Action	Purpose
Step 1	Create an IPv6 ACL, and enter IPv6 access list configuration mode.	

	Command or Action	Purpose
Step 2	Configure the IPv6 ACL to block (deny) or pass (permit) traffic.	
Step 3	Apply the IPv6 ACL to the interface where the traffic needs to be filtered.	
Step 4	Apply the IPv6 ACL to an interface. For router ACLs, you must also configure an IPv6 address on the Layer 3 interface to which the ACL is applied.	

Default IPv6 ACL Configuration

There are no IPv6 ACLs configured or applied.

Interaction 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, for any additional configured ACLs, packets are dropped to the CPU, and the ACLs are applied in software. When the hardware is full a message is printed to the console indicating the ACL has been unloaded and the packets will be dropped on the interface.

Note   Only packets of the same type as the ACL that could not be added (ipv4, ipv6, MAC) will be dropped on the interface.

How To Configure an IPv6 ACL

Creating IPv6 ACL

Beginning in privileged EXEC mode, follow these steps to create an IPv6 ACL:

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 access-list** *acl_name*
3. **{deny|permit} protocol**
4. **{deny|permit} tcp**
5. **{deny|permit} udp**
6. **{deny|permit} icmp**
7. **end**
8. **show ipv6 access-list**
9. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# `**`configure terminal`**	Enters global configuration mode.
Step 2	**ipv6 access-list** *acl_name* **Example:** `ipv6 access-list access-list-name`	Use a name to define an IPv6 access list and enter IPv6 access-list configuration mode.
Step 3	**{deny|permit} protocol** **Example:** `{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]`	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: • For protocol, enter the 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. • The source-ipv6-prefix/prefix-length or destination-ipv6-prefix/ prefix-length 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 source-ipv6-address or destination-ipv6-address, 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.

Command or Action	Purpose
	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. • (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. • (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 4294967295 • (Optional) Enter time-range name to specify the time range that applies to the deny or permit statement.
Step 4 **{deny|permit} tcp** **Example:** `{deny	permit} tcp {source-ipv6-prefix/prefix-length` `| any	hostsource-ipv6-address}` `[operator` `[port-number]]{destination-ipv6-prefix/prefix-length` `| any |hostdestination-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 3, 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}—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.

Command or Action	Purpose
	• syn—Synchronize bit set.
	• urg—Urgent pointer bit set.
Step 5 **{deny|permit} udp** **Example:** `{deny	permit} udp {source-ipv6-prefix/prefix-length` `| any	hostsource-ipv6-address}` `[operator` `[port-number]]{destination-ipv6-prefix/prefix-length` `| any	hostdestination-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 6 **{deny|permit} icmp** **Example:** `{deny	permit} icmp` `{source-ipv6-prefix/prefix-length	any |` `hostsource-ipv6-address}` `[operator [port-number]]` `{destination-ipv6-prefix/prefix-length	any |` `hostdestination-ipv6-address}` `[operator [port-number]][icmp-type [icmp-code]` `|icmp-message] [dscpvalue] [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 or see command reference for this release.
Step 7 **end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.
Step 8 **show ipv6 access-list** **Example:** `show ipv6 access-list`	Verify the access list configuration.
Step 9 **copy running-config startup-config** **Example:** `copy running-config startup-config`	(Optional) Save your entries in the configuration file.

Related Topics

Applying an IPv6 to an Interface

This section describes how to apply IPv6 ACLs to network interfaces. You can apply an IPv6 ACL to outbound or inbound traffic on layer 2 and Layer 3 interfaces. You can apply IPv6 ACLs only to inbound management traffic on Layer 3 interfaces.

Beginning in privileged EXEC mode, follow these steps to control access to an interface:

SUMMARY STEPS

1. **configure terminal**
2. **interface** interface_id
3. **no switchport**
4. **ipv6 address** *ipv6_address*
5. **ipv6 traffic-filter** *acl_name*
6. **end**
7. **show running-config interface tenGigabitEthernet 1/0/3**
8. **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# configure terminal`	Enters global configuration mode.
Step 2	**interface** interface_id **Example:** `Switch# interface interface-id`	Identifies a Layer 2 interface (for port ACLs) or Layer 3 Switch Virtual interface (for router ACLs) on which to apply an access list, and enters interface configuration mode.
Step 3	**no switchport** **Example:** `Switch# no switchport`	Changes the interface from Layer 2 mode (the default) to Layer 3 mode (only if applying a router ACL).

	Command or Action	Purpose
Step 4	**ipv6 address** *ipv6_address* **Example:** `Switch# ipv6 address ipv6-address`	Configures 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 has already been configured with an explicit IPv6 address.
Step 5	**ipv6 traffic-filter** *acl_name* **Example:** `Switch# ipv6 traffic-filter access-list-name {in	out}`	Applies the access list to incoming or outgoing traffic on the interface.
Step 6	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Step 7	**show running-config interface tenGigabitEthernet 1/0/3**
	
	Example:
	`Switch# show running-config interface`
	`tenGigabitEthernet 1/0/3`
	`.........................`
	`.........................`
	`Building configuration ............`
	`.........................`
	`Current configuration : 98 bytes`
	`!`
	`interface TenGigabitEthernet1/0/3`
	` switchport mode trunk`
	` ipv6 traffic-filter MyFilter out`
	`end`
		Shows the configuration summary.

Step 8	**copy running-config startup-config** **Example:** `copy running-config startup-config`	(Optional) Saves your entries in the configuration file.

Related Topics

Creating WLAN IPv6 ACL

SUMMARY STEPS

1. **ipv6 traffic-filter acl** *acl_name*
2. **ipv6 traffic-filter acl web**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 traffic-filter acl** *acl_name* **Example:** `Switch(config-wlan)# ipv6 traffic-filter acl <acl_name>`	Creates a named WLAN ACL.
Step 2	**ipv6 traffic-filter acl web** **Example:** `Switch(config-wlan)# ipv6 traffic-filter acl web <acl_name-preauth>`	Creates a pre-authentication for WLAN ACL.

```
Switch(config-wlan)# ipv6 traffic-filter acl <acl_name>
Switch(config-wlan)#ipv6 traffic-filter acl web <acl_name-preauth>
```

Related Topics

Verifying IPv6 ACL

Displaying 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.

DETAILED STEPS

	Command or Action	Purpose
Step 1	**show access-list** **Example:** `Switch# show access-lists`	Displays all access lists configured on the switch
Step 2	**show ipv6 access-list** *acl_name* **Example:** `Switch# show ipv6 access-list [access-list-name]`	Displays all configured IPv6 access list or the access list specified by name.

Related Topics

Configuration Examples for IPv6 ACL

Example: Creating 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.

```
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: 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.

```
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-lists 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
```

Example: Configuring RA Throttling and NS Suppression

This task describes how to create an RA throttle policy in order to help the power-saving wireless clients from being disturbed by frequent unsolicited periodic RA's. The unsolicited multicast RA is throttled by the controller.

Before You Begin

Enable IPv6 on the client machine.

SUMMARY STEPS

1. **configure terminal**
2. **ipv6 nd ra-throttler policy** Mythrottle
3. **throttle-period** 20
4. **max-through** 5
5. **allow at-least** 3 **at-most** 5
6. **switch (config)# vlan configuration 100**
7. **ipv6 nd suppress**
8. **ipv6 nd ra-th attach-policy** attach-policy_name
9. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**configure terminal** **Example:** `Switch# `**`configure terminal`**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	**ipv6 nd ra-throttler policy** Mythrottle **Example:** `Switch (config)# ipv6 nd ra-throttler policy Mythrottle`	Creates a RA throttler policy called Mythrottle.
Step 3	**throttle-period** 20 **Example:** `Switch (config-nd-ra-throttle)# throttle-period 20`	Determines the time interval segment during which throttling applies.
Step 4	**max-through** 5 **Example:** `Switch (config-nd-ra-throttle)# max-through 5`	Determines how many initial RA's are allowed.
Step 5	**allow at-least** 3 **at-most** 5 **Example:** `Switch (config-nd-ra-throttle)# allow at-least 3 at-most 5`	Determines how many RA's are allowed after the initial RAs have been transmitted, until the end of the interval segment.
Step 6	**switch (config)# vlan configuration 100** **Example:** `Switch (config)# vlan configuration 100`	Creates a per vlan configuration.
Step 7	**ipv6 nd suppress** **Example:** `Switch (config)# ipv6 nd suppress`	Disables the neighbor discovery on the Vlan.
Step 8	**ipv6 nd ra-th attach-policy** attach-policy_name **Example:** `Switch (config)# ipv6 nd ra-throttle attach-policy attach-policy_name`	Enables the router advertisement throttling.
Step 9	**end** **Example:** `Switch(config)# end`	Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Example: Configuring RA Guard Policy

SUMMARY STEPS

1. **ipv6 nd raguard policy** MyPloicy
2. **trusted-port**
3. **device-role router**
4. **interface tenGigabitEthernet** 1/0/1
5. **ipv6 nd raguard attach-policy**MyPolicy
6. **vlan configuration** 19-21,23
7. **ipv6 nd suppress**
8. **ipv6 snooping**
9. **ipv6 nd raguard attach-policy** MyPolicy
10. **ipv6 nd ra-throttler attach-policy** Mythrottle

DETAILED STEPS

	Command or Action	Purpose
Step 1	**ipv6 nd raguard policy** MyPloicy **Example:** `Switch (config)# ipv6 nd raguard policy MyPolicy`	
Step 2	**trusted-port** **Example:** `Switch (config-nd-raguard)# trusted-port`	Configures the trusted port for the policy created above.

Step 3	**device-role router**
	
	Example:
	`Switch (config-nd-raguard)# device-role`
	`[host|monitor|router|switch]`
	`Switch (config-nd-raguard)# device-role router`
		Defines the trusted device that can send RAs to the trusted port created above.

Step 4	**interface tenGigabitEthernet** 1/0/1 **Example:** `Switch (config)# interface tenGigabitEthernet 1/0/1`	Configures the interface to the trusted device.

Step 5	**ipv6 nd raguard attach-policy**MyPolicy
	
	Example:
	`Switch (config-if)# ipv6 nd raguard attach-policy`
	`Mypolicy`
		Configures and attaches the policy to trust the RA's received from the port.

	Command or Action	Purpose
Step 6	**vlan configuration** 19-21,23 **Example:** `Switch (config)# vlan configuration 19-21,23`	Configures the wireless client vlans.
Step 7	**ipv6 nd suppress** **Example:** `Switch (config-vlan-config)# ipv6 nd suppress`	Suppresses the ND messages over wireless.
Step 8	**ipv6 snooping** **Example:** `Switch (config-vlan-config)# ipv6 snooping`	Captures IPv6 traffic.

Step 9	**ipv6 nd raguard attach-policy** MyPolicy
	
	Example:
	`Switch (config-vlan-config)# ipv6 nd raguard`
	`attach-policy Mypolicy`
		Attaches the RA Guard policy to the wireless client vlans.

Step 10	**ipv6 nd ra-throttler attach-policy** Mythrottle
	
	Example:
	`Switch (config-vlan-config)#ipv6 nd ra-throttler`
	`attach-policy Mythrottle`
		Attaches the RA throttling policy to the wireless client vlans.

Example: Configuring IPv6 Neighbor Binding

SUMMARY STEPS

1. **ipv6 neighbor binding** [vlan]19 2001:db8::25:4 **interface tenGigabitEthernet** 1/0/3 aaa.bbb.ccc

DETAILED STEPS

	Command or Action	Purpose

Step 1	**ipv6 neighbor binding** [vlan]19 2001:db8::25:4 **interface tenGigabitEthernet** 1/0/3 aaa.bbb.ccc
	
	Example:
	`Switch (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

Related Topic	Document Title
IPv6 command reference	*IPv6 Command Reference (Catalyst 3650 Switches)*
ACL configuration	*Security Configuration Guide (Catalyst 3650 Switches)*

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 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
The Cisco Support website provides extensive online resources, including documentation and tools for troubleshooting and resolving technical issues with Cisco products and technologies. 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. Access to most tools on the Cisco Support website requires a Cisco.com user ID and password.	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 3.2SECisco IOS XE 3.3SE	This feature was introduced.