Configuring IP ACLs

This chapter describes how to configure IP access control lists (ACLs) on Cisco NX-OS .

Unless otherwise specified, the term IP ACL refers to IPv4 ACLs.

Note The Cisco NX-OS release that is running on a managed may not support all documented features or settings. For the latest feature information and caveats, see the documentation and release notes for your platform and software release.

This chapter includes the following sections:

Information About ACLs

An ACL is an ordered set of rules that you can use to filter traffic. Each rule specifies a set of conditions that a packet must satisfy to match the rule. When the device determines that an ACL applies to a packet, it tests the packet against the conditions of all rules. The first matching rule determines whether the packet is permitted or denied. If there is no match, the device applies the applicable implicit rule. The device continues processing packets that are permitted and drops packets that are denied.

You can use ACLs to protect networks and specific hosts from unnecessary or unwanted traffic. For example, you could use ACLs to disallow HTTP traffic from a high-security network to the Internet. You could also use ACLs to allow HTTP traffic but only to specific sites, using the IP address of the site to identify it in an IP ACL.

ACL Types and Applications

The device supports the following types of ACLs for security traffic filtering:

IPv4 ACLs

The device applies IPv4 ACLs only to IPv4 traffic.

IP ACLs have the following types of applications:

Port ACL

Filters Layer 2 traffic

Router ACL

Filters Layer 3 traffic

VLAN ACL

Filters VLAN traffic

This table summarizes the applications for security ACLs.

Table 1: Security ACL Applications

Application	Supported Interfaces	Types of ACLs Supported
Port ACL	• Layer 2 interfaces • Layer 2 Ethernet port-channel interfaces When a port ACL is applied to a trunk port, the ACL filters traffic on all VLANs on the trunk port.	• IPv4 ACLs

Application	Supported Interfaces	Types of ACLs Supported
Router ACL	• VLAN interfaces • Physical Layer 3 interfaces • Layer 3 Ethernet subinterfaces • Layer 3 Ethernet port-channel interfaces • Layer 3 Ethernet port-channel subinterfaces • Management interfaces **Note** You must enable VLAN interfaces globally before you can configure a VLAN interface.	• IPv4 ACLs
VLAN ACL	• VLANs	• IPv4 ACLs

Order of ACL Application

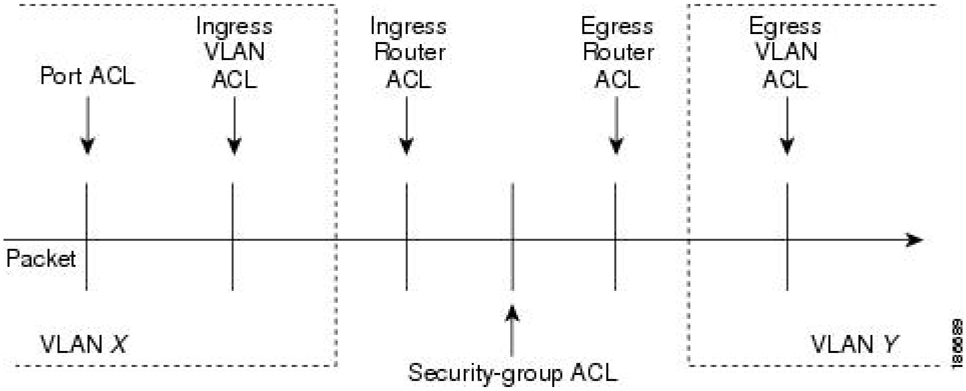
When the device processes a packet, it determines the forwarding path of the packet. The path determines which ACLs that the device applies to the traffic. The device applies the ACLs in the following order:

1. Port ACL
2. Ingress VACL
3. Ingress router ACL
4. Egress router ACL
5. Egress VACL

If the packet is bridged within the ingress VLAN, the device does not apply router ACLs.

The following figure shows the order in which the device applies ACLs.

Figure 1: Order of ACL Application



The following figure shows where the device applies ACLs, depending upon the type of ACL. The red path indicates a packet sent to a destination on a different interface than its source. The blue path indicates a packet that is bridged within its VLAN.

The device applies only the applicable ACLs. For example, if the ingress port is a Layer 2 port and the traffic is on a VLAN that is a VLAN interface, a port ACL and a router ACL both can apply. In addition, if a VACL is applied to the VLAN, the device applies that ACL too.

Figure 2: ACLs and Packet Flow

About Rules

Rules are what you create, modify, and remove when you configure how an ACL filters network traffic. Rules appear in the running configuration. When you apply an ACL to an interface or change a rule within an ACL that is already applied to an interface, the supervisor module creates ACL entries from the rules in the running configuration and sends those ACL entries to the applicable I/O module. Depending upon how you configure the ACL, there may be more ACL entries than rules, especially if you implement policy-based ACLs by using object groups when you configure rules.

You can create rules in access-list configuration mode by using the **permit** or **deny** command. The device allows traffic that matches the criteria in a permit rule and blocks traffic that matches the criteria in a deny rule. You have many options for configuring the criteria that traffic must meet in order to match the rule.

This section describes some of the options that you can use when you configure a rule. For information about every option, see the applicable **permit** and **deny** commands in the *Cisco Nexus 7000 Series NX-OS Security Command Reference*.

Protocols for IP ACLs

IPv4 ACLs allow you to identify traffic by protocol. For your convenience, you can specify some protocols by name. For example, in an IPv4 ACL, you can specify ICMP by name.

You can specify any protocol by number.

In IPv4, you can specify protocols by the integer that represents the Internet protocol number. For example, you can use 115 to specify Layer 2 Tunneling Protocol (L2TP) traffic.

For a list of the protocols that each type of ACL supports by name, see the applicable **permit** and **deny** commands in the *Cisco Nexus 3000 Series NX-OS Command Reference.*.

Source and Destination

In each rule, you specify the source and the destination of the traffic that matches the rule. You can specify both the source and destination as a specific host, a network or group of hosts, or any host. How you specify the source and destination depends on whether you are configuring IPv4 ACLs.

ACL TCAM Regions

You can change the size of the ACL ternary content addressable memory (TCAM) regions in the hardware.

The IPv4 TCAMs are single wide. However, the IPv6 TCAMs are double wide. For example, to create a 256-entry IPv6 TCAM, you need to reduce a IPv4 TCAM by 256 x 2, or 512 entries.

You can create IPv6 port ACLs, VLAN ACL, router ACLs, and you can match IPv6 addresses for QoS. However, Cisco NX-OS cannot support all of them simultaneously. You must remove or reduce the size of the existing TCAMs to enable these new IPv6 TCAMs.

TCAM region sizes have the following guidelines and limitations:

- To revert to the default ACL TCAM size, use the **no hardware profile tcam region** command. You no longer need to use the **write erase command** and reload the switch.
- Depending upon the platform, each TCAM region might have a different minimum/maximum/aggregate size restriction.
- The default size of the ARPACL TCAM is zero. Before you use the ARP ACLs in a Control Policing Plane (CoPP) policy, you must set the size of this TCAM to a non-zero size.
- You must set the VACL and egress VLAN ACL (E-VACL) size to the same value.
- Both IPv4 and IPv6 addresses cannot coexist, even in a double-wide TCAM.
- The total TCAM depth is 2000 for ingress and 1000 for egress, which can be carved in 256 entries blocks.
- After TCAM carving, you must reload the switch.
- All existing TCAMs cannot be set to size 0.
- By default, all IPv6 TCAMs are disabled (the TCAM size is set to 0).

Table 2: TCAM Sizes by ACL Region

TCAM ACL Region	Default Size	Minimum Size	Incremental Size	Maximum Size
SUP (ingress)	128 x 2	128 x 2	N/A	128 x 2
SPAN (ingress)	128	128	N/A	128
ARPACL (ingress)	0	0	128	128

TCAM ACL Region	Default Size	Minimum Size	Incremental Size	Maximum Size
PACL (ingress)	384	ARPACL disabled = 128 ARPACL enabled = 256	256	1664 (combined)
VACL (ingress)	512	0	256	
RACL (ingress)	512	256	256	
QOS (ingress)	256	256	256	
PACL_IPV6 (ingress)	0	0	256 x 2	
VACL_IPV6 (ingress)	0	0	256 x 2	
RACL_IPV6 (ingress)	0	0	256 x 2	
QOS_IPV6 (ingress)	0	0	256 x 2	
E-VACL (egress)	512	0	256	1024 (combined)
E-RACL (egress)	512	0	256	
E-VACL_IPV6 (egress)	0	0	256 x 2	
E-RACL_IPV6 (egress)	0	0	256 x 2	
QOSLBL (pre-lookup)	256	256	256	512(combined)
IPSG (pre-lookup)	256	256	256	
SUP_IPV6 (pre-lookup)	128 x 2	256 x 2	N/A	256 x 2

Implicit Rules for IP ACLs

IP ACLs have implicit rules, which means that although these rules do not appear in the running configuration, the device applies them to traffic when no other rules in an ACL match. When you configure the device to maintain per-rule statistics for an ACL, the device does not maintain statistics for implicit rules.

All IPv4 ACLs include the following implicit rule:

```
deny ip any any
```

This implicit rule ensures that the device denies unmatched IP traffic.

This implicit rule ensures that the device denies the unmatched traffic, regardless of the protocol specified in the Layer 2 header of the traffic.

Additional Filtering Options

You can identify traffic by using additional options. These options differ by ACL type. The following list includes most but not all additional filtering options:

- IPv4 ACLs support the following additional filtering options:
    - Layer 4 protocol
    - Authentication Header Protocol
    - Enhanced Interior Gateway Routing Protocol (EIGRP)
    - Open Shortest Path First (OSPF)
    - Payload Compression Protocol
    - Protocol-independent multicast (PIM)
    - TCP and UDP ports
    - ICMP types and codes
    - Precedence level
    - Differentiated Services Code Point (DSCP) value
    - TCP packets with the ACK, FIN, PSH, RST, SYN, or URG bit set
    - Established TCP connections
    - Packet length

Sequence Numbers

The device supports sequence numbers for rules. Every rule that you enter receives a sequence number, either assigned by you or assigned automatically by the device. Sequence numbers simplify the following ACL tasks:

Adding new rules between existing rules

By specifying the sequence number, you specify where in the ACL a new rule should be positioned. For example, if you need to insert a rule between rules numbered 100 and 110, you could assign a sequence number of 105 to the new rule.

Removing a rule

Without using a sequence number, removing a rule requires that you enter the whole rule, as follows:

```
switch(config-acl)# no permit tcp 10.0.0.0/8 any
```

However, if the same rule had a sequence number of 101, removing the rule requires only the following command:

```
switch(config-acl)# no 101
```

Moving a rule

With sequence numbers, if you need to move a rule to a different position within an ACL, you can add a second instance of the rule using the sequence number that positions it correctly, and then you can remove the original instance of the rule. This action allows you to move the rule without disrupting traffic.

If you enter a rule without a sequence number, the device adds the rule to the end of the ACL and assigns a sequence number that is 10 greater than the sequence number of the preceding rule to the rule. For example, if the last rule in an ACL has a sequence number of 225 and you add a rule without a sequence number, the device assigns the sequence number 235 to the new rule.

In addition, Cisco NX-OS allows you to reassign sequence numbers to rules in an ACL. Resequencing is useful when an ACL has rules numbered contiguously, such as 100 and 101, and you need to insert one or more rules between those rules.

Logical Operators and Logical Operation Units

IP ACL rules for TCP and UDP traffic can use logical operators to filter traffic based on port numbers. The device stores operator-operand couples in registers called logical operator units (LOUs).

The LOU usage for each type of operator is as follows:

eq

Is never stored in an LOU

gt

Uses 1 LOU

lt

Uses 1 LOU

neq

Uses 1 LOU

range

Uses 1 LOU

Statistics and ACLs

The device can maintain global statistics for each rule that you configure in IPv4 and IPv6 ACLs. If an ACL is applied to multiple interfaces, the maintained rule statistics are the sum of packet matches (hits) on all the interfaces on which that ACL is applied.

Note The device does not support interface-level ACL statistics.

For each ACL that you configure, you can specify whether the device maintains statistics for that ACL, which allows you to turn ACL statistics on or off as needed to monitor traffic filtered by an ACL or to help troubleshoot the configuration of an ACL.

The device does not maintain statistics for implicit rules in an ACL. For example, the device does not maintain a count of packets that match the implicit **deny ip any any** rule at the end of all IPv4 ACLs. If you want to maintain statistics for implicit rules, you must explicitly configure the ACL with rules that are identical to the implicit rules.

Session Manager Support for IP ACLs

Session Manager supports the configuration of IP. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration.

Licensing Requirements for IP ACLs

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	No license is required to use IP ACLs. However to support up to 128K ACL entries using an XL line card, you must install the scalable services license. Any feature not included in a license package is bundled with the Cisco NX-OS system images and is provided at no extra charge to you. For an explanation of the Cisco NX-OS licensing scheme, see the *Cisco NX-OS Licensing Guide*.

Prerequisites for IP ACLs

IP ACLs have the following prerequisites:

- You must be familiar with IP addressing and protocols to configure IP ACLs.
- You must be familiar with the interface types that you want to configure with ACLs.

Guidelines and Limitations for IP ACLs

IP ACLs have the following configuration guidelines and limitations:

- We recommend that you perform ACL configuration using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. This is especially useful for ACLs that include more than about 1000 rules.
- Packets that fail the Layer 3 maximum transmission unit check and therefore require fragmenting.
- IPv4 packets that have IP options (additional IP packet header fields following the destination address field).
- When you apply an ACL that uses time ranges, the device updates the ACL entries whenever a time range referenced in an ACL entry starts or ends. Updates that are initiated by time ranges occur on a best-effort priority. If the device is especially busy when a time range causes an update, the device may delay the update by up to a few seconds.
- To apply an IP ACL to a VLAN interface, you must have enabled VLAN interfaces globally.

Default Settings for IP ACLs

This table lists the default settings for IP ACL parameters.

Table 3: Default IP ACL Parameters

Parameters	Default
IP ACLs	No IP ACLs exist by default
ACL rules	Implicit rules apply to all ACLs
Object groups	No object groups exist by default
Time ranges	No time ranges exist by default

Configuring IP ACLs

Creating an IP ACL

You can create an IPv4 ACL ACL on the device and add rules to it.

Before You Begin

We recommend that you perform the ACL configuration using the Session Manager. This feature allows you to verify the ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. This feature is especially useful for ACLs that include more than about 1000 rules.

SUMMARY STEPS

1. **configure terminal**
2. Enter one of the following commands:
   - **ip access-list** *name*
3. (Optional) **fragments** {**permit-all** | **deny-all**}
4. [*sequence-number*] {**permit** | **deny**} *protocol source destination*
5. (Optional) **statistics per-entry**
6. (Optional) Enter one of the following commands:
   - **show ip access-lists** *name*
7. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	Enter one of the following commands:
	• **ip access-list** *name*
	Example:
	`switch(config)# ip access-list acl-01`
	`switch(config-acl)#`
		Creates the IP ACL and enters IP ACL configuration mode. The *name* argument can be up to 64 characters.

Step 3	**fragments** {**permit-all** | **deny-all**}
	Example:
	`switch(config-acl)# fragments permit-all`
		(Optional)
		Optimizes fragment handling for noninitial fragments. When a device applies to traffic an ACL that contains the **fragments** command, the **fragments** command only matches noninitial fragments that do not match any explicit **permit** or **deny** commands in the ACL.

Step 4	[*sequence-number*] {**permit** | **deny**} *protocol source destination* **Example:** `switch(config-acl)# permit ip 192.168.2.0/24 any`	Creates a rule in the IP ACL. You can create many rules. The *sequence-number* argument can be a whole number between 1 and 4294967295. The **permit** and **deny** commands support many ways of identifying traffic.

	Command or Action	Purpose
Step 5	**statistics per-entry** **Example:** `switch(config-acl)# statistics per-entry`	(Optional) Specifies that the device maintains global statistics for packets that match the rules in the ACL.
Step 6	Enter one of the following commands: • **show ip access-lists** *name* **Example:** `switch(config-acl)# show ip access-lists acl-01`	(Optional) Displays the IP ACL configuration.
Step 7	**copy running-config startup-config** **Example:** `switch(config-acl)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Changing an IP ACL

You can add and remove rules in an existing IPv4 ACL, but you cannot change existing rules. Instead, to change a rule, you can remove it and recreate it with the desired changes.

If you need to add more rules between existing rules than the current sequence numbering allows, you can use the **resequence** command to reassign sequence numbers.

Before You Begin

We recommend that you perform ACL configuration using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. This feature is especially useful for ACLs that include more than about 1000 rules.

SUMMARY STEPS

1. **configure terminal**
2. Enter one of the following commands:
   - **ip access-list** *name*
3. (Optional) [*sequence-number*] {**permit** | **deny**} *protocol source destination*
4. (Optional) [**no**] **fragments** {**permit-all** | **deny-all**}
5. (Optional) **no** {*sequence-number* | {**permit** | **deny**} *protocol source destination*}
6. (Optional) [**no**] **statistics per-entry**
7. (Optional) Enter one of the following commands:
   - **show ip access-lists** *name*
8. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	Enter one of the following commands:
	• **ip access-list** *name*
	Example:
	`switch(config)# ip access-list acl-01`
	`switch(config-acl)#`
		Enters IP ACL configuration mode for the ACL that you specify by name.

Step 3	[*sequence-number*] {**permit** | **deny**} *protocol source destination* **Example:** `switch(config-acl)# 100 permit ip 192.168.2.0/24 any`	(Optional) Creates a rule in the IP ACL. Using a sequence number allows you to specify a position for the rule in the ACL. Without a sequence number, the rule is added to the end of the rules. The *sequence-number* argument can be a whole number between 1 and 4294967295. The **permit** and **deny** commands support many ways of identifying traffic.

Step 4	[**no**] **fragments** {**permit-all** | **deny-all**}
	Example:
	`switch(config-acl)# fragments permit-all`
		(Optional)
		Optimizes fragment handling for noninitial fragments. When a device applies to traffic an ACL that contains the **fragments** command, the **fragments** command only matches noninitial fragments that do not match any explicit **permit** or **deny** commands in the ACL.

	Command or Action	Purpose
		The **no** option removes fragment-handling optimization.

Step 5	**no** {*sequence-number* | {**permit** | **deny**} *protocol source destination*}
	Example:
	`switch(config-acl)# no 80`
		(Optional)
		Removes the rule that you specified from the IP ACL.
		The **permit** and **deny** commands support many ways of identifying traffic.

Step 6	[**no**] **statistics per-entry** **Example:** `switch(config-acl)# statistics per-entry`	(Optional) Specifies that the device maintains global statistics for packets that match the rules in the ACL. The **no** option stops the device from maintaining global statistics for the ACL.
Step 7	Enter one of the following commands: • **show ip access-lists** *name* **Example:** `switch(config-acl)# show ip access-lists acl-01`	(Optional) Displays the IP ACL configuration.
Step 8	**copy running-config startup-config** **Example:** `switch(config-acl)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Changing Sequence Numbers in an IP ACL

You can change all the sequence numbers assigned to the rules in an IP ACL.

Before You Begin

We recommend that you perform ACL configuration using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources required by the configuration are available prior to committing them to the running configuration. This feature is especially useful for ACLs that include more than about 1000 rules.

SUMMARY STEPS

1. **configure terminal**
2. (Optional) **show ip access-lists** *name*
3. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	**show ip access-lists** *name* **Example:** `switch(config)# show ip access-lists acl-01`	(Optional) Displays the IP ACL configuration.
Step 3	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Removing an IP ACL

You can remove an IP ACL from the device.

Before You Begin

Ensure that you know whether the ACL is applied to an interface. The device allows you to remove ACLs that are currently applied. Removing an ACL does not affect the configuration of interfaces where you have applied the ACL. Instead, the device considers the removed ACL to be empty. Use the **show ip access-lists** command command with the summary keyword to find the interfaces that an IP ACL is configured on.

SUMMARY STEPS

1. **configure terminal**
2. Enter one of the following commands:
   - **no ip access-list** *name*
3. (Optional) Enter one of the following commands:
   - **show ip access-lists** *name* **summary**
4. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	Enter one of the following commands: • **no ip access-list** *name* **Example:** `switch(config)# no ip access-list acl-01`	Removes the IP ACL that you specified by name from the running configuration.
Step 3	Enter one of the following commands: • **show ip access-lists** *name* **summary** **Example:** `switch(config)# show ip access-lists acl-01 summary`	(Optional) Displays the IP ACL configuration. If the ACL remains applied to an interface, the command lists the interfaces.
Step 4	**copy running-config startup-config** **Example:** `switch(config)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Applying an IP ACL as a Router ACL

You can apply an IPv4 or IPv6 ACL to any of the following types of interfaces:

- Physical Layer 3 interfaces and subinterfaces
- Layer 3 Ethernet port-channel interfaces and subinterfaces
- VLAN interfaces
- Tunnels
- Management interfaces

ACLs applied to these interface types are considered router ACLs.

Note Logical operation units (LOUs) are not available for router ACLs applied in the out direction. If an IPv4 ACL is applied as a router ACL in the out direction, access control entiries (ACEs) that contain logical operators for TCP/UDP port numbers are expanded internally to multiple ACEs and might require more TCAM entries when compared to the same ACL applied in the in direction.

Before You Begin

Ensure that the ACL you want to apply exists and that it is configured to filter traffic in the manner that you need for this application.

SUMMARY STEPS

1. **configure terminal**
2. Enter one of the following commands:
   - **interface ethernet** *slot*/*port*[**.** *number*]
   - **interface port-channel** *channel-number*[**.** *number*]
   - **interface tunnel** *tunnel-number*
   - **interface vlan** *vlan-ID*
   - **interface mgmt** *port*
3. Enter one of the following commands:
   - **ip access-group** *access-list* {**in** | **out**}
   - **ipv6 traffic-filter** *access-list* {**in** | **out**}
4. (Optional) **show running-config aclmgr**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	Enter one of the following commands: • **interface ethernet** *slot*/*port*[**.** *number*] • **interface port-channel** *channel-number*[**.** *number*] • **interface tunnel** *tunnel-number* • **interface vlan** *vlan-ID*	Enters configuration mode for the interface type that you specified.

	Command or Action	Purpose

	- **interface mgmt** *port*
	Example:
	`switch(config)# interface tunnel 13`
	`switch(config-if)#`	

Step 3	Enter one of the following commands: - **ip access-group** *access-list* {**in** | **out**} - **ipv6 traffic-filter** *access-list* {**in** | **out**} **Example:** `switch(config-if)# ip access-group acl-120 out`	Applies an IPv4 or IPv6 ACL to the Layer 3 interface for traffic flowing in the direction specified. You can apply one router ACL per direction.
Step 4	**show running-config aclmgr** **Example:** `switch(config-if)# show running-config aclmgr`	(Optional) Displays the ACL configuration.
Step 5	**copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Applying an IP ACL as a Port ACL

You can apply an IPv4 or IPv6 ACL to a Layer 2 interface, which can be a physical port or a port channel. ACLs applied to these interface types are considered port ACLs.

Before You Begin

Ensure that the ACL you want to apply exists and that it is configured to filter traffic in the manner that you need for this application.

Note If the interface is configured with the **mac packet-classify** command, you cannot apply an IP port ACL to the interface until you remove the **mac packet-classify** command from the interface configuration.

SUMMARY STEPS

1. **configure terminal**
2. Enter one of the following commands:
   - **interface ethernet** *slot*/*port*
   - **interface port-channel** *channel-number*
3. Enter one of the following commands:
   - **ip port access-group** *access-list* **in**
   - **ipv6 port traffic-filter** *access-list* **in**
4. (Optional) **show running-config aclmgr**
5. (Optional) **copy running-config startup-config**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config)#`
		Enters global configuration mode.

Step 2	Enter one of the following commands:
	• **interface ethernet** *slot*/*port*
	• **interface port-channel** *channel-number*
	Example:
	`switch(config)# interface ethernet 2/3`
	`switch(config-if)#`
		Enters configuration mode for the interface type that you specified.

Step 3	Enter one of the following commands:
	• **ip port access-group** *access-list* **in**
	• **ipv6 port traffic-filter** *access-list* **in**
	Example:
	`switch(config-if)# ip port access-group`
	`acl-l2-marketing-group in`
		Applies an IPv4 or IPv6 ACL to the interface or port channel. Only inbound filtering is supported with port ACLs. You can apply one port ACL to an interface.

Step 4	**show running-config aclmgr** **Example:** `switch(config-if)# show running-config aclmgr`	(Optional) Displays the ACL configuration.

	Command or Action	Purpose
Step 5	**copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	(Optional) Copies the running configuration to the startup configuration.

Applying an IP ACL as a VACL

You can apply an IP ACL as a VACL.

Configuring ACL TCAM Region Sizes

You can change the size of the ACL ternary content addressable memory (TCAM) regions in the hardware.

SUMMARY STEPS

1. **configure terminal**
2. **hardware profile tcam region** {**arpacl** | {**ipv6-e-racl** | **e-racl**} | **ifacl** | **ipsg** | {**ipv6-qos** | **qos**} |**qoslbl** | {**ipv6-racl** | **racl**} | **vacl** } *tcam_size*
3. **copy running-config startup-config**
4. switch(config)# **show hardware profile tcam region**
5. switch(config)# **reload**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config) #`
		Enters global configuration mode.

Step 2	**hardware profile tcam region** {**arpacl** | {**ipv6-e-racl** | **e-racl**} | **ifacl** | **ipsg** | {**ipv6-qos** | **qos**} |**qoslbl** | {**ipv6-racl** | **racl**} | **vacl** } *tcam_size*	Changes the ACL TCAM region size. • **arpacl**—Configures the size of the Address Resolution Protocol (ARP) ACL (ARPACL) TCAM region. • **e-racl**—Configures the size of the egress router ACL (ERACL) TCAM region. • **e-vacl**—Configures the size of the egress VLAN ACL (EVACL) TCAM region. • **ifacl**—Configures the size of the interface ACL (ifacl) TCAM region.

	Command or Action	Purpose
		• **ipsg**—Configures the size of the IP Source Guard (IPSG) TCAM region. • **qos**—Configures the size of the quality of service (QoS) TCAM region. • **qoslbl**—Configures the size of the QoS Label (qoslbl) TCAM region. • **racl**—Configures the size of the router ACL (RACL) TCAM region. • **vacl**—Configures the size of the VLAN ACL (VACL) TCAM region. • *tcam_size*—TCAM size. The range is from 0 to 2,14,74, 83, 647 entries. **Note** **vacl** and **e-vacl** TCAM regions should be set to the same size.
Step 3	**copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 4	switch(config)# **show hardware profile tcam region**	Displays the TCAM sizes that will be applicable on the next reload of the switch.
Step 5	switch(config)# **reload**	Copies the running configuration to the startup configuration. **Note** The new size values are effective only upon the next reload after saving the **copy running-config to startup-config**.

The following example shows how to change the size of the RACL TCAM region:

```
switch(config)# hardware profile tcam region racl 256
[SUCCESS] New tcam size will be applicable only at boot time.
You need to 'copy run start' and 'reload'

switch(config)# copy running-configur startup-config
switch(config)# reload
WARNING: This command will reboot the system
Do you want to continue? (y/n) [n] y
```

The following example shows the error message you see when you set the ARP ACL TCAM value to a value other than 0 or 128, and then shows how to change the size of the ARP ACL TCAM region and verify the changes:

```
switch(config)# hardware profile tcam region arpacl 200
ARPACL size can be either 0 or 128

switch(config)# hardware profile tcam region arpacl 128
To start using ARPACL tcam, IFACL tcam size needs to be changed.
Changing IFACL tcam size to 256
[SUCCESS] New tcam size will be applicable only at boot time.
You need to 'copy run start' and 'reload'

switch(config)# show hardware profile tcam region
   sup size = 128
  vacl size = 512
 ifacl size = 256
   qos size = 256
 rbacl size = 0
  span size = 128
  racl size = 256
e-racl size = 512
e-vacl size = 512
qoslbl size = 512
```

```
  ipsg size = 512
arpacl size = 128
switch(config)#
```

The following example shows how to configure the TCAM VLAN ACLs on a switch:

```
switch# configure sync
Enter configuration commands, one per line. End with CNTL/Z.
switch(config-sync)# switch-profile s5010
Switch-Profile started, Profile ID is 1
switch(config-sync-sp)# hardware profile tcam region vacl 512
switch(config-sync-sp)# hardware profile tcam region e-vacl 512
switch(config-sync-sp)#
```

Reverting to the Default TCAM Region Sizes

SUMMARY STEPS

1. **configure terminal**
2. switch(config)# **no hardware profile tcam region** { **arpacl** | **arpacl** *tcam_size*}
3. (Optional) **copy running-config startup-config**
4. switch(config)# **reload**

DETAILED STEPS

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`switch# configure terminal`
	`switch(config) #`
		Enters global configuration mode.

Step 2	switch(config)# **no hardware profile tcam region** { **arpacl** | **arpacl** *tcam_size*}	Reverts the configuration to the default ACL TCAM size.

Step 3	**copy running-config startup-config**
	Example:
	`switch(config) # copy running-config-startup-config`
	`switch(config) #`
		(Optional)
		Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Step 4	switch(config)# **reload**	Reloads the switch.

The following example shows how to revert to the default RACL TCAM region sizes:

```
switch(config)# no hardware profile tcam region racl 256
[SUCCESS] New tcam size will be applicable only at boot time.
You need to 'copy run start' and 'reload'

switch(config)# copy running-configur startup-config
switch(config)# reload
WARNING: This command will reboot the system
Do you want to continue? (y/n) [n] y
```

Verifying the IP ACL Configuration

To display IP ACL configuration information, perform one of the following tasks.

Command	Purpose
show running-config aclmgr [**all**]	Displays the ACL running configuration, including the IP ACL configuration and the interfaces to which IP ACLs are applied.
show startup-config aclmgr [**all**]	Displays the ACL startup configuration.

Monitoring and Clearing IP ACL Statistics

To monitor or clear IP ACL statistics, use one of the commands in this table.

Command	Purpose
show ip access-lists	Displays the IPv4 ACL configuration. If the IPv4 ACL includes the **statistics per-entry** command, the **show ip access-lists** command output includes the number of packets that have matched each rule.
clear ip access-list counters	Clears statistics for all IPv4 ACLs or for a specific IPv4 ACL.

Configuration Examples for IP ACLs

The following example shows how to create an IPv4 ACL named acl-01 and apply it as a port ACL to Ethernet interface 2/1, which is a Layer 2 interface:

```
ip access-list acl-01
  permit ip 192.168.2.0/24 any
interface ethernet 2/1
  ip port access-group acl-01 in
```

The following example shows how to create an IPv6 ACL named acl-120 and apply it as a router ACL to Ethernet interface 2/3, which is a Layer 3 interface:

```
ipv6 access-list acl-120
  permit tcp 2001:0db8:85a3::/48 2001:0db8:be03:2112::/64
  permit udp 2001:0db8:85a3::/48 2001:0db8:be03:2112::/64
  permit tcp 2001:0db8:69f2::/48 2001:0db8:be03:2112::/64
  permit udp 2001:0db8:69f2::/48 2001:0db8:be03:2112::/64
interface ethernet 2/3
```

```
ipv6 traffic-filter acl-120 in
```