Configuring IP ACLs

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

Unless otherwise specified, the term IP ACL refers to IPv4 and IPv6 ACLs.

Information About ACLs

An access control list (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 switch determines that an ACL applies to a packet, it tests the packet against the conditions of all rules. The first match determines whether the packet is permitted or denied. If there is no match, the switch applies the applicable default rule. The switch 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.

IP ACL Types and Applications

The Cisco Nexus device supports IPv4, IPv6, and MAC ACLs for security traffic filtering. The switch allows you to use IP access control lists (ACLs) as port ACLs, and Router ACLs as shown in the following table.

Table 1: Security ACL Applications

Application	Supported Interfaces	Types of ACLs Supported
Port ACL	An ACL is considered a port ACL when you apply it to one of the following: • Ethernet interface • Ethernet port-channel interface	IPv4 ACLs IPv6 ACLs MAC ACLs
Router ACL	• Physical Layer 3 interfaces • Layer 3 Ethernet subinterfaces • Layer 3 Ethernet port-channel interfaces • Layer 3 Ethernet port-channel subinterfaces • Management interfaces • Switched Virtual Interfaces (SVIs)	IPv4 ACLs IPv6 ACLs
VTY ACL	VTYs	IPv4 ACLs IPv6 ACLs

Application Order

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 Router ACL

Rules

You can create rules in access-list configuration mode by using the **permit** or **deny** command. The switch 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.

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.

Protocols

IPv4, IPv6, and MAC 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 the integer that represents the Internet protocol number.

Implicit Rules

IP and MAC ACLs have implicit rules, which means that although these rules do not appear in the running configuration, the switch applies them to traffic when no other rules in an ACL match.

All IPv4 ACLs include the following implicit rule:

```
deny ip any any
```

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

All IPv6 ACLs include the following implicit rule:

```
deny ipv6 any any
```

All MAC ACLs include the following implicit rule:

```
deny any any protocol
```

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. IPv4 ACLs support the following additional filtering options:

- Layer 4 protocol
- TCP and UDP ports
- IGMP types
- Established TCP connections

Sequence Numbers

The Cisco Nexus 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, the device 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 Cisco Nexus device stores operator-operand couples in registers called logical operation units (LOUs) to perform operations (greater than, less than, not equal to, and range) on the TCP and UDP ports specified in an IP ACL.

Licensing Requirements for ACLs

The following table shows the licensing requirements for this feature:

Product	License Requirement
Cisco NX-OS	No license is required to use ACLs.

Prerequisites for 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 ACLs

IP ACLs have the following configuration guidelines and limitations:

- As an enhancement to HTTP method match, the tcp-option-length option has been added to the ACE syntax to specify the length of the TCP options header in the packets. You can configure up to four tcp-option-lengths in the ACEs, which include the TCP option length of 0. If you do not configure the tcp-option-length option, the length is considered as 0. It means that only the packets without the TCP options header can match this ACE. This feature gives more flexibility in such a way that the HTTP method can be matched even on the packets that have the variable length TCP options header.
- We recommend that you perform ACL configuration using the Session Manager. This feature allows you to verify ACL configuration and confirm that the resources that are required by the configuration are available before committing them to the running configuration. This is especially useful for ACLs that include more than about 1000 rules.

- You can configure any number of ACLs as long as TCAM space is available.
- Egress RACLs are not supported in Release 7.x. although the configuration may be allowed without an error or warning.
- Usually, ACL processing for IP packets occurs on the I/O modules, which use hardware that accelerates ACL processing. In some circumstances, processing occurs on the supervisor module, which can result in slower ACL processing, especially during processing that involves an ACL with many rules. Management interface traffic is always processed on the supervisor module. If IP packets in any of the following categories are exiting a Layer 3 interface, they are sent to the supervisor module for processing:
  - 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).
  - IPv6 packets that have extended IPv6 header fields.
- When you apply an ACL that uses time ranges, the device updates the ACL entries whenever a time range that is 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. Make sure that the time range is valid and in an active state.
- To use the **match-local-traffic** option for all inbound and outbound traffic, you must first enable the ACL in the software.
- For a Cisco N3K-C36180YC-R switch with configured egress RACLs, before upgrading from a 7.x release to a 9.x release, follow these steps to ensure the RACLs are maintained and the upgrade is completed without issue:
  1. Add TCAM entries for egress RACL using the **hardware access-list tcam region e-racl** command.
  2. Save the configuration and reload.
  3. Upgrade to a 9.x release.

  For more information about configuring TCAM regions, see ACL TCAM Regions, on page 15 and Configuring ACL TCAM Region Sizes, on page 22.
- In Cisco Nexus Release 9.2(2), Cisco Nexus 3636C-R and 36180YC-R switches do not support the following on egress RACL:
  - UDF to support ICMP Type Match
  - ACL log on egress
  - Egress IPv4 RACL with additional filter option tcp/udp ports with lt/gt
  - Egress IPv4 RACL with additional filter option tcp/udp ports with neq
  - Egress IPv4 RACL with additional filter option tcp/udp ports with range
  - Egress IPv4 RACL with flag
  - Egress RACL on an external TCAM
  - Egress PACL support

- Statistics support
- Label sharing

- In Cisco NX-OS Release 9.2(3), Cisco Nexus 3636C-R and 36180YC-R switches support the following on the ACLs:
  - Statistics support
  - Label sharing
- In Cisco NX-OS Release 9.2(3), Cisco Nexus 3636C-R and 36180YC-R switches have the following guidelines:
  - Atomic ACL update is supported for all the ingress ACL features except for the Multihop BFD and CoPP features.
  - Atomic ACL update is not supported for the egress ACL features.
  - Label sharing is supported only for the same policy on different interfaces within the same ASIC.
  - In Cisco NX-OS Release 9.2(3), ACL statistics are not supported for the following:
    - BFD
    - DHCP - IPv4 and IPv6
    - PACL-MAC
    - PACL- IPv6
    - PBR - IPv4 and IPv6
    - RACL-IPv6
    - RACL-IPv4 when using an external TCAM
- When you enable the counters for the ACL TCAM entries using the hardware profile acl-stats module xx command, the input discard field in the show interface is always zero. This limitation is applicable only to the Cisco Nexus 3600 platform switches with N3K-C3636C-R and N3K-C36180YC-R line cards.

Default ACL Settings

The following table lists the default settings for IP ACLs parameters.

Table 2: Default IP ACLs Parameters

Parameters	Default
IP ACLs	No IP ACLs exist by default.
ACL rules	Implicit rules apply to all ACLs .

The following table lists the default settings for MAC ACLs parameters.

Table 3: Default MAC ACLs Parameters

Parameters	Default
MAC ACLs	No MAC ACLs exist by default.
ACL rules	Implicit rules apply to all ACLs .

Configuring IP ACLs

Creating an IP ACL

You can create an IPv4 or IPv6 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 that are required by the configuration are available before committing them to the running configuration. This feature is especially useful for ACLs that include more than about 1000 rules.

Procedure

	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*
	• **ipv6 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	[*sequence-number*] {**permit** | **deny**} *protocol* {*source-ip-prefix* | *source-ip-mask*} {*destination-ip-prefix* | *destination-ip-mask*} **Example:** `switch(config-acl)# 10 permit ipv6 1::1 2::2 3::3 4::4`	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. For more information, see the *Command Reference* for the specific Cisco Nexus device.

	Command or Action	Purpose
Step 4	**statistics per-entry** **Example:** `switch(config-acl)# statistics per-entry`	Specifies that the device maintains global statistics for packets that match the rules in the ACL.
Step 5	**hardware profile acl-stats module** *xx* **Example:** `switch(config-acl)# hardware profile acl-stats module 1`	Enables counters for the ACL TCAM entries on both, the internal and external TCAM. **Note** This command is applicable only for Cisco Nexus 9500 platform switches with -R and -RX line cards and Cisco Nexus 3636C-R and 36180YC-R switches. VLAN and SVI statistics are lost when you enable the counters.
Step 6	**reload** **Example:** `switch(config)# reload`	Reloads the switch. **Note** The **reload** command is mandatory for the Cisco Nexus 3636C-R and 36180YC-R switches.
Step 7	Enter one of the following commands: • **show ip access-lists** *name* • **show ipv6 access-lists** *name* **Example:** `switch(config-acl)# show ip access-lists acl-01`	Displays the IP ACL configuration.
Step 8	**copy running-config startup-config** **Example:** `switch(config-acl)# copy running-config startup-config`	Copies the running configuration to the startup configuration.
Step 9	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Example

This example shows how to create an IPv4 ACL:

```
switch# configure terminal
switch(config)# ip access-list acl-01
switch(config-acl)# permit ip 192.168.2.0/24 any
```

This example shows how to create an IPv6 ACL:

```
switch# configure terminal
switch(config)# ipv6 access-list acl-01-ipv6
switch(config-ipv6-acl)# permit tcp 2001:0db8:85a3::/48 2001:0db8:be03:2112::/64
```

Configuring IPv4 ACL Logging

To configure the IPv4 ACL logging process, you first create the access list, then enable filtering of IPv4 traffic on an interface using the specified ACL, and finally configure the ACL logging process parameters.

Procedure

	Command or Action	Purpose

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

Step 2	**ip access-list** *name*
	Example:
	`switch(config)# ip access-list logging-test`
	`switch(config-acl)#`
		Creates an IPv4 ACL and enters IP ACL configuration mode. The *name* argument can be up to 64 characters.

Step 3	{**permit** | **deny**} **ip** *source-address destination-address* **log** **Example:** `switch(config-acl)# permit ip any 10.30.30.0/24 log`	Creates an ACL rule that permits or denies IPv4 traffic matching its conditions. To enable the system to generate an informational logging message about each packet that matches the rule, you must include the **log** keyword. The *source-address* and *destination-address* arguments can be the IP address with a network wildcard, the IP address and variable-length subnet mask, the host address, or **any** to designate any address.

Step 4	**exit**
	Example:
	`switch(config-acl)# exit`
	`switch(config)#`
		Updates the configuration and exits IP ACL configuration mode.

Step 5	**interface ethernet** *slot/port*
	Example:
	`switch(config)# interface ethernet 1/1`
	`switch(config-if)#`
		Enters interface configuration mode.

Step 6	**ip access-group** *name* **in** **Example:** `switch(config-if)# ip access-group logging-test in`	Enables the filtering of IPv4 traffic on an interface using the specified ACL. You can apply an ACL to inbound traffic.
Step 7	**exit** **Example:**	Updates the configuration and exits interface configuration mode.

	Command or Action	Purpose
	`switch(config-if)# exit` `switch(config)#`	
Step 8	**logging ip access-list cache interval** *interval* **Example:** `switch(config)# logging ip access-list cache interval 490`	Configures the log-update interval (in seconds) for the ACL logging process. The default value is 300 seconds. The range is from 5 to 86400 seconds.
Step 9	**logging ip access-list cache entries** *number-of-flows* **Example:** `switch(config)# logging ip access-list cache entries 8001`	Specifies the maximum number of flows to be monitored by the ACL logging process. The default value is 8000. The range of values supported is from 0 to 1048576.
Step 10	**logging ip access-list cache threshold** *threshold* **Example:** `switch(config)# logging ip access-list cache threshold 490`	If the specified number of packets is logged before the expiry of the alert interval, the system generates a syslog message.
Step 11	**logging ip access-list detailed** **Example:** `switch(config)# logging ip access-list detailed`	Enables the ACL name, the sequence number of ACE, action, ACL direction, ACL filter type, and the ACL applied interface are displayed in the output of the **show logging ip access-list cache** command.
Step 12	**hardware rate-limiter access-list-log** *packets* **Example:** `switch(config)# hardware rate-limiter access-list-log 200`	Configures rate limits in packets per second for packets copied to the supervisor module for ACL logging. The range is from 0 to 30000. **Note** Cisco Nexus NX-OS 7.0(3)F3(1) does not support the **hardware rate-limiter access-list-log** command.
Step 13	**acllog match-log-level** *severity-level* **Example:** `switch(config)# acllog match-log-level 5`	Specifies the minimum severity level to log ACL matches. The default is 6 (informational). The range is from 0 (emergency) to 7 (debugging).
Step 14	(Optional) **show logging ip access-list cache** [**detail**] **Example:** `switch(config)# show logging ip access-list cache`	Displays information on the active logged flows, such as source IP and destination IP addresses, source port and destination port information, source interfaces, and so on.

Changing an IP ACL

You can add and remove rules in an existing IPv4 or IPv6 ACL. 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.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# {**ip** | **ipv6**}**ip access-list** *name*	Enters IP ACL configuration mode for the ACL that you specify by name.
Step 3	switch(config)# **ip access-list** *name*	Enters IP ACL configuration mode for the ACL that you specify by name.
Step 4	switch(config-acl)# [*sequence-number*] {**permit** | **deny**} *protocol source destination*	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. For more information, see the *Command Reference* for your Cisco Nexus device.
Step 5	(Optional) switch(config-acl)# **no** {*sequence-number* | {**permit** | **deny**} *protocol source destination*}	Removes the rule that you specified from the IP ACL. The **permit** and **deny** commands support many ways of identifying traffic. For more information, see the *Command Reference* for your Cisco Nexus device.
Step 6	(Optional) switch#**show ip access-lists** *name*	Displays the IP ACL configuration.
Step 7	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Related Topics

Changing Sequence Numbers in an IP ACL, on page 12

Removing an IP ACL

You can remove an IP ACL from the switch.

Before you remove an IP ACL from the switch, be sure that you know whether the ACL is applied to an interface. The switch 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 switch considers the removed ACL to be empty.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **no** {**ip** | **ipv6**} **access-list** *name*	Removes the IP ACL that you specified by name from the running configuration.
Step 3	switch(config)# no **ip access-list** *name*	Removes the IP ACL that you specified by name from the running configuration.
Step 4	(Optional) switch# **show running-config**	Displays the ACL configuration. The removed IP ACL should not appear.
Step 5	(Optional) switch# **copy running-config startup-config**	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.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	(Optional) switch# **show** {**ip** | **ipv6**} **access-lists** *name*	Displays the IP ACL configuration.
Step 3	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Applying an IP ACL to mgmt0

You can apply an IPv4 or IPv6 ACL to the management interface (mgmt0).

Before you begin

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

Procedure

	Command or Action	Purpose

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

Step 2	**ip access-group** *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 3	(Optional) **show running-config aclmgr** **Example:** `switch(config-if)# show running-config aclmgr`	Displays the ACL configuration.
Step 4	(Optional) **copy running-config startup-config** **Example:** `switch(config-if)# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Related Topics

- Creating an IP ACL

Applying an IP ACL as a Port ACL

You can apply an IPv4 ACL to a physical Ethernet interface or a PortChannel. ACLs applied to these interface types are considered port ACLs.

Note Some configuration parameters when applied to an PortChannel are not reflected on the configuration of the member ports.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface** {**ethernet** [*chassis*/]*slot*/*port* | **port-channel** *channel-number*}	Enters interface configuration mode for the specified interface.
Step 3	(Optional) switch# **show running-config**	Displays the ACL configuration.

	Command or Action	Purpose
Step 4	(Optional) switch# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Applying an IP ACL as a Router ACL

You can apply an IPv4 or 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
- Management interfaces

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.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	Enter one of the following commands: • switch(config)# **interface ethernet** *slot*/*port*[**.** *number*] • switch(config)# **interface port-channel** *channel-number*[**.** *number*] • switch(config)# **interface mgmt** *port*	Enters configuration mode for the interface type that you specified.
Step 3	Enter one of the following commands: • switch(config-if)# **ip access-group** *access-list* {*in*} • switch(config-if)# **ipv6 traffic-filter** *access-list* {*in*}	Applies an IPv4 or IPv6 ACL to the layer 3 interface for traffic in the ingress direction.
Step 4	(Optional) switch(config-if)# **show running-config aclmgr**	Displays the ACL configuration.
Step 5	(Optional) switch(config-if)# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

About System ACLs

You can configure system ACLs on Cisco Nexus 36180YC-R and C3636C-R switches. With system ACLs, you can now configure a Layer 2 port ACL (PACL) on all the ports with the same access-list in the switch.

Configuring system ACLs reduces the TCAM usage and also brings down the time and memory usage while the policy is being applied or modified.

See the following guidelines and limitations for configuring system ACLs:

- The system PACL is supported for Layer 2 interface only.
- ACE statistics are not yet supported for the system ACLs.
- IPv6 is not yet supported in the system ACLs.
- System ACLs are not supported on the breakout port.
- For quality of service, ACL, or TCAM carving configuration on Cisco Nexus 3600 platform switches, see the Cisco Nexus 3600 NX-OS Quality of Service Configuration Guide for more information.

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.

You can create IPv6 port ACLs, router ACLs, and you can match IPv6 addresses for QoS. Cisco NX-OS provides simultaneous support for all three TCAMs. 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 access list tcam region** command. You need to reload the modules when you revert to default sizes.
- Depending upon the platform, each TCAM region might have a different minimum/maximum/aggregate size restriction.
- The total number of TCAMs is 16.
  - There are 12 large TCAMs—Each has 2048 entries that are 160 bit key size.
  - There are 4 small TCAMs—Each has 256 entries that are 160 bit key size.
- The TCAM regions RACL v6, QoS, CoPP, and Multicast cannot be set to 0.
- Redirect_v6, RACL v4 cannot share TCAM with any other features.
- After TCAM carving, you must reload the switch.
- RACL v6, CoPP, and multicast have default TCAM sizes and these TCAM sizes must be non-zero on the following Cisco 3600 line cards to avoid line card failure during reload:
  - N3K-C3636C-R
  - N3K-C36180YC-R
- You can partially use IPv6 RACL with IPv6 IFCAL. This is applicable Cisco Nexus N3K-C36180YC-R and N3K-C3636C-R line cards.

Table 4: TCAM Sizes by ACL Region

TCAM ACL Region	Default Size
PACL_IPv4 [ifacl]	1024
PACL_IPV6 [ipv6-ifacl]	1024
PACL_MAC [mac-ifacl]	2048
IPv4 Port QOS [qos]	640
IPv6 Port QOS [ipv6-qos]	256
IPv4 RACL [racl]	1024
IPv6 RACL [ipv6-racl]	1024
IPv4 L3 QoS [l3qos]	640
IPv6 L3 QoS [ipv6-l3qos]	256
SPAN [span]	96
Ingress COPP [copp]	128
Redirect v4	1024
Redirect v6	2048

Carving a TCAM Region

Before configuring the system ACLs, carve the TCAM region first. Note that for configuring the ACLs less than 1k, you do not need to carve the TCAM region. See the Configuring ACL TCAM Region Sizes, on page 22 section for more information.

Note You can configure PACL IPv4, RACL IPv4, and RACL IPv6 beyond 12k.

Configuring System ACLs

After an IPv4 ACL is created, configure the system ACL.

Before you begin

Create an IPv4 ACL on the device. See Creating an IP ACL, on page 7 for more information.

Procedure

	Command or Action	Purpose
Step 1	**config t**	Enters the configuration mode.

	Command or Action	Purpose
Step 2	**system acl**	Configures the system ACL.
Step 3	**ip port access-group** <*pacl name*> **in**	Applies a Layer 2 PACL to the interface. Only inbound filtering is supported with port ACLs. You can apply one port ACL to an interface.

Configuration and Show Command Examples for the System ACLs

See the following configuration examples for the system ACL show commands.

Configuring system PACL with 1K scale [using default TCAM]

See the following example for configuring system PACL with 1K scale [Using default TCAM].

Step 1: Create PACL.

```
config t
ip access-list PACL-DNA
    10 permit ip 1.1.1.1/32 any
    20 permit tcp 3.0.0.0/8 255.0.0.0 eq 1500
    25 deny udp any any eq 500
    26 deny tcp any eq 490 any
    …·· ….

    1000 deny any any
```

Step 2: Apply PACL into system level.

```
configuration terminal
system acl
    ip port access-group PACL-DNA in
```

To validate the system ACLs that are configured on the switch, use the **sh run aclmgr | sec system** command:

```
switch# sh run aclmgr | sec system
system acl
 ip port access-group test in
switch#
```

To validate the PACLs that are configured on the switch, use the **sh ip access-lists <name> [summary]** command:

```
switch# sh ip access-lists test

IP access list test
        10 deny udp any any eq 27
        20 permit ip 1.1.1.1/32 100.100.100.100/32
        30 permit ip 1.2.1.1/32 100.100.100.100/32
        40 permit ip 1.3.1.1/32 100.100.100.100/32
        50 permit ip 1.4.1.1/32 100.100.100.100/32
        60 permit ip 1.5.1.1/32 100.100.100.100/32
        70 permit ip 1.6.1.1/32 100.100.100.100/32
        80 permit ip 1.7.1.1/32 100.100.100.100/32
        90 permit ip 1.8.1.1/32 100.100.100.100/32
```

```
switch# sh ip access-lists test summary
IPV4 ACL test
        Total ACEs Configured: 12279
        Configured on interfaces:
        Active on interfaces:
                - ingress
                - ingress

switch#
```

To validate PACL IPv4 (ifacl) TCAM region size, use the **show hardware access-list tcam region** command:

```
switch# show hardware access-list tcam region
*********************************WARNING*********************************
*****************The output shows NFE tcam region info******************
***Please refer to 'show hardware access-list tcam template' for NFE2***
*************************************************************************
                              IPV4 PACL [ifacl] size = 12280
                         IPV6 PACL [ipv6-ifacl] size =    0
                           MAC PACL [mac-ifacl] size =    0
                            IPV4 Port QoS [qos] size =  640
                       IPV6 Port QoS [ipv6-qos] size =  256
                         MAC Port QoS [mac-qos] size =    0
                      FEX IPV4 PACL [fex-ifacl] size =    0
                 FEX IPV6 PACL [fex-ipv6-ifacl] size =    0
                   FEX MAC PACL [fex-mac-ifacl] size =    0
                    FEX IPV4 Port QoS [fex-qos] size =    0
               FEX IPV6 Port QoS [fex-ipv6-qos] size =    0
                 FEX MAC Port QoS [fex-mac-qos] size =    0
                               IPV4 VACL [vacl] size =    0
                          IPV6 VACL [ipv6-vacl] size =    0
                            MAC VACL [mac-vacl] size =    0
                           IPV4 VLAN QoS [vqos] size =    0
                      IPV6 VLAN QoS [ipv6-vqos] size =    0
                        MAC VLAN QoS [mac-vqos] size =    0
                               IPV4 RACL [racl] size =    0
                            IPV6 RACL [ipv6-racl] size =  128
                  IPV4 Port QoS Lite [qos-lite] size =    0
           FEX IPV4 Port QoS Lite [fex-qos-lite] size =    0
                 IPV4 VLAN QoS Lite [vqos-lite] size =    0
                  IPV4 L3 QoS Lite [l3qos-lite] size =    0
                        Egress IPV4 QoS [e-qos] size =    0
                   Egress IPV6 QoS [e-ipv6-qos] size =    0
                     Egress MAC QoS [e-mac-qos] size =    0
                        Egress IPV4 VACL [vacl] size =    0
                   Egress IPV6 VACL [ipv6-vacl] size =    0
                     Egress MAC VACL [mac-vacl] size =    0
                      Egress IPV4 RACL [e-racl] size =    0
                 Egress IPV6 RACL [e-ipv6-racl] size =    0
             Egress IPV4 QoS Lite [e-qos-lite] size =    0
                               IPV4 L3 QoS [l3qos] size =  640
                          IPV6 L3 QoS [ipv6-l3qos] size =  256
                            MAC L3 QoS [mac-l3qos] size =    0
                                    Ingress System size =    0
                                     Egress System size =    0
                                       SPAN [span] size =   96
                                  Ingress COPP [copp] size =  128
                        Ingress Flow Counters [flow] size =    0
switch#
```

To view ACL related tech support information, use the **show tech-support aclmgr** and **show tech-support aclqos** commands.

```
show tech-support aclmgr
show tech-support aclqos
```

Configuring ACL Logging

ACL Logging

The Cisco Nexus device supports ACL logging, which allows you to monitor flows that hit specific access control lists (ACLs). To enable the feature for the ACL entry, configure specific ACEs with the optional **log** keyword.

Configuring the ACL Logging Cache

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **logging ip access-list cache entries** *num_entries*	Sets the maximum number of log entries cached in the software. The range is from 0 to 1000000 entries. The default value is 8000 entries.
Step 3	switch(config)# **logging ip access-list cache interval** *seconds*	Sets the number of seconds between log updates. If an entry is inactive for this duration, it is removed from the cache. The range is from 5 to 86400 seconds. The default value is 300 seconds.
Step 4	switch(config)# **logging ip access-list cache threshold** *num_packets*	Sets the number of packet matches before an entry is logged. The range is from 0 to 1000000 packets. The default value is 0 packets, which means that logging is not triggered by the number of packet matches.
Step 5	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example show how to set the maximum number of log entries to 5000, the interval to 120 seconds, and the threshold to 500000:

```
switch# configure terminal
switch(config)# logging ip access-list cache entries 5000
```

```
switch(config)# logging ip access-list cache interval 120
switch(config)# logging ip access-list cache threshold 500000
switch(config)# copy running-config startup-config
```

Applying ACL Logging to an Interface

You can apply ACL logging to Ethernet interfaces and port channels.

Before you begin

- Create an ACL.
- Create an IP access list with at least one access control entry (ACE) configured for logging.
- Configure the ACL logging cache.
- Configure the ACL log match level.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **interface ethernet** *slot/port*	Specifies the Ethernet interface.
Step 3	switch(config-if)# **ip access-group** *name* **in**	Attaches an ACL with a log to the specified interface. ACL logging is enabled when the ACL is applied to the interface on the hardware.
Step 4	(Optional) switch(config-if)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to apply the Ethernet interface with the logging specified in acl1 for all ingress traffic:

```
switch# configure terminal
switch(config)# interface ethernet 1/2
switch(config-if)# ip access-group acl1 in
switch(config-if)# copy running-config startup-config
```

Applying the ACL Log Match Level

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

	Command or Action	Purpose
Step 2	switch(config)# **acllog match-log-level** *number*	Specifies the logging level to match for entries to be logged in the ACL log (acllog). The number is a value from 0 to 7. The default is 6. **Note** Log messages are entered into the log if the logging level for the ACL log facility (acllog) and the logging severity level for the log file are greater than or equal to the ACL log match log level setting.
Step 3	(Optional) switch(config)# **copy running-config startup-config**	Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.

Example

The following example shows how to apply the log match level for entries to be logged in the ACL log:

```
switch# configure terminal
switch(config)# acllog match-log-level 3
switch(config)# copy running-config startup-config
```

Clearing Log Files

You can clear messages in the log file and the NVRAM.

Procedure

	Command or Action	Purpose
Step 1	switch# **clear logging ip access-list cache**	Clears the access control list (ACL) cache.

Verifying the ACL Logging Configuration

To display ACL logging configuration information, perform one of the following tasks:

Command	Purpose
switch# **show hardware access-list tcam region**	Displays the TCAM sizes that will be applicable on the next reload of the device.
switch# **show ip access-lists**	Displays the IPv4 ACL configuration.
switch# **show ipv6 access-lists**	Displays the IPv6 ACL configuration.

Command	Purpose
switch# **show logging ip access-list cache [detail]**	Displays information on the active logged flows, such as source IP and destination IP addresses, source port and destination port information, and source interfaces.
switch# **show logging ip access-list status**	Displays the deny maximum flow count, the current effective log interval, and the current effective threshold value..
switch# **show startup-config acllog**	Displays the access control list (ACL) log file in the startup configuration.
switch# **show startup-config aclmgr [all]**	Displays the access control list (ACL) log file in the startup configuration. **Note** This command displays the user-configured ACLs in the startup configuration. The all option displays both the default (CoPP-configured) and the user-configured ACLs in the startup configuration.
switch# **show running-config acllog**	Displays the access control list (ACL) log file in the running configuration.
switch# **show running-config aclmgr [all]**	Displays the access control list (ACL) log file in the running configuration including the IP ACL configuration and the interfaces where you have applied IP ACLs. **Note** This command displays the user-configured ACLs in the running configuration. The all option displays both the default (CoPP-configured) and the user-configured ACLs in the startup configuration.

Configuring ACL TCAM Region Sizes

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

Note You cannot change the size of the small TCAMs (TCAM 12 through 15)

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	**hardware access-list tcam region** { **ifacl** | {**ipv6-qos** | **qos**} | {**ipv6-racl** | **racl**} *tcam_size*	Changes the ACL TCAM region size. • **ifacl**—Configures the size of the interface ACL (ifacl) TCAM region. The maximum number of entries is 1500. • **qos**—Configures the size of the quality of service (QoS) TCAM region. • **racl**—Configures the size of the router ACL (RACL) TCAM region. • *tcam_size*—TCAM size. The range is from 0 to 256, 512, (multiples of 256) entries.
Step 3	**copy running-config startup-config** **Example:** `switch(config)# 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 access-list tcam region** **Example:** `switch(config)# show hardware access-list tcam region`	Displays the TCAM sizes that will be applicable on the next reload of the switch.
Step 5	switch(config)# **reload** **Example:** `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**.

Example

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

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

This example shows how to display the TCAM region sizes to verify your changes:

```
switch(config)# show hardware accesslist tcam region | exclude "0"


        IPV4 PACL [ifacl] size = 1024
   IPV6 PACL [ipv6-ifacl] size = 1024
     MAC PACL [mac-ifacl] size = 2048
      IPV4 Port QoS [qos] size =  640
 IPV6 Port QoS [ipv6-qos] size =  256
         IPV4 RACL [racl] size = 2048
    IPV6 RACL [ipv6-racl] size = 1024
      IPV4 L3 QoS [l3qos] size =  640
 IPV6 L3 QoS [ipv6-l3qos] size =  256
              SPAN [span] size =   96
      Ingress COPP [copp] size =  128
              Redirect v4 size =  1024
              Redirect v6 size =  2048
```

Reverting to the Default TCAM Region Sizes

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **no hardware profile tcam region** {**arpacl** | **e-racl**} | **ifacl** | **ipsg** | **qos**} |**qoslbl** | **racl**} | **vacl** } *tcam_size*	Reverts the configuration to the default ACL TCAM size.
Step 3	(Optional) switch(config)# **copy running-config startup-config**	Saves the changes persistently through reboots and restarts by copying the running configuration to the startup configuration.
Step 4	switch(config)# **reload**	Reloads the switch.

Example

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
```

Configuring ACLs on Virtual Terminal Lines

To restrict incoming and outgoing connections for IPv4 or IPv6 between a Virtual Terminal (VTY) line and the addresses in an access list, use the **access-class** command in line configuration mode. To remove access restrictions, use the **no** form of this command.

Follow these guidelines when configuring ACLs on VTY lines:

- Set identical restrictions on all VTY lines because a user can connect to any of them.
- Statistics per entry is not supported for ACLs on VTY lines.

Before you begin

Be sure that the ACL that you want to apply exists and is configured to filter traffic for this application.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.

Step 2	switch(config)# **line vty**
	Example:
	`switch(config)# line vty`
	`switch(config-line)#`
		Enters line configuration mode.

Step 3	switch(config-line)# **access-class access-list-number {in | out}**
	Example:
	`switch(config-line)# access-class ozi2 in`
	`switch(config-line)#access-class ozi3 out`
	`switch(config)#`
		Specifies inbound or outbound access restrictions.

Step 4	(Optional) switch(config-line)# **no access-class access-list-number {in | out}**
	Example:
	`switch(config-line)# no access-class ozi2 in`
	`switch(config-line)# no access-class ozi3 out`
	`switch(config)#`
		Removes inbound or outbound access restrictions.

Step 5	switch(config-line)# **exit**
	Example:
	`switch(config-line)# exit`
	`switch#`
		Exits line configuration mode.

	Command or Action	Purpose
Step 6	(Optional) switch# **show running-config aclmgr** **Example:** `switch# show running-config aclmgr`	Displays the running configuration of the ACLs on the switch.
Step 7	(Optional) switch# **copy running-config startup-config** **Example:** `switch# copy running-config startup-config`	Copies the running configuration to the startup configuration.

Example

The following example shows how to apply the access-class ozi2 command to the in-direction of the vty line.

```
switch# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
switch(config)# line vty
switch(config-line)# access-class ozi2 in
switch(config-line)# exit
switch#
```

Verifying ACLs on VTY Lines

To display the ACL configurations on VTY lines, perform one of the following tasks:

Command	Purpose
show running-config aclmgr	Displays the running configuration of the ACLs configured on the switch.
show users	Displays the users that are connected.
show access-lists *access-list-name*	Display the statistics per entry.

Configuration Examples for ACLs on VTY Lines

The following example shows the connected users on the console line (ttyS0) and the VTY lines (pts/0 and pts/1).

```
switch# show users
NAME     LINE         TIME         IDLE          PID COMMENT
admin    ttyS0        Aug 27 20:45   .         14425 *
admin    pts/0        Aug 27 20:06 00:46       14176 (172.18.217.82) session=ssh
admin    pts/1        Aug 27 20:52   .         14584 (10.55.144.118)
```

The following example shows how to allow vty connections to all IPv4 hosts except 172.18.217.82 and how to deny vty connections to any IPv4 host except 10.55.144.118, 172.18.217.79, 172.18.217.82, 172.18.217.92:

```
switch# show running-config aclmgr
!Time: Fri Aug 27 22:01:09 2010
version 5.0(2)N1(1)
ip access-list ozi
  10 deny ip 172.18.217.82/32 any
  20 permit ip any any
ip access-list ozi2
  10 permit ip 10.55.144.118/32 any
  20 permit ip 172.18.217.79/32 any
  30 permit ip 172.18.217.82/32 any
  40 permit ip 172.18.217.92/32 any


line vty
  access-class ozi in
  access-class ozi2 out
```

The following example shows how to configure the IP access list by enabling per-entry statistics for the ACL:

```
switch# configure terminal
Enter configuration commands, one per line.
End with CNTL/Z.
switch(config)# ip access-list ozi2
switch(config-acl)# statistics per-entry
switch(config-acl)# deny tcp 172.18.217.83/32 any
switch(config-acl)# exit

switch(config)# ip access-list ozi
switch(config-acl)# statistics per-entry
switch(config-acl)# permit ip 172.18.217.20/24 any
switch(config-acl)# exit
switch#
```

The following example shows how to apply the ACLs on VTY in and out directions:

```
switch(config)# line vty
switch(config-line)# ip access-class ozi in
switch(config-line)# access-class ozi2 out
switch(config-line)# exit
switch#
```

The following example shows how to remove the access restrictions on the VTY line:

```
switch# configure terminal
Enter configuration commands, one per line. End
with CNTL/Z.
switch(config)# line vty
switch(config-line)# no access-class ozi2 in
switch(config-line)# no ip access-class ozi2 in
switch(config-line)# exit
switch#
```