Configuring Access Control Lists

This chapter contains the following sections:

- Information About ACLs, on page 1
- Configuring IP ACLs, on page 9
- Configuring ACL Using HTTP Methods to Redirect Requests, on page 19
- Information About VLAN ACLs, on page 21
- Configuring VACLs, on page 21
- Configuration Examples for VACL, on page 24
- Configuring the LOU Threshold, on page 24
- Configuring ACL TCAM Region Sizes, on page 25
- Configuring ACLs on Virtual Terminal Lines, on page 28

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 for security traffic filtering. The switch allows you to use IP access control lists (ACLs) as port ACLs, VLAN 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 When a port ACL is applied to a trunk port, the ACL filters traffic on all VLANs on the trunk port.	IPv4 ACLs IPv6 ACLs
Router ACL	• VLAN interfaces **Note** You must enable VLAN interfaces globally before you can configure a VLAN interface. • Physical Layer 3 interfaces • Layer 3 Ethernet subinterfaces • Layer 3 Ethernet port-channel interfaces • Layer 3 Ethernet port-channel subinterfaces • Tunnels • Management interfaces	IPv4 ACLs IPv6 ACLs
VLAN ACL (VACL)	An ACL is a VACL when you use an access map to associate the ACL with an action and then apply the map to a VLAN.	IPv4 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 VACL
3. Ingress Router ACL
4. Egress Router ACL
5. Egress VACL

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


permit icmp any any nd-na
permit icmp any any nd-ns
permit icmp any any router-advertisement
permit icmp any any router-solicitation
```

Unless you configure an IPv6 ACL with a rule that denies ICMPv6 neighbor discovery messages, the first four rules ensure that the device permits neighbor discovery advertisement and solicitation messages. The fifth rule ensures that the device denies unmatched IPv6 traffic.

Note If you explicitly configure an IPv6 ACL with a **deny ipv6 any any** rule, the implicit permit rules can never permit traffic. If you explicitly configure a **deny ipv6 any any** rule but want to permit ICMPv6 neighbor discovery messages, explicitly configure a rule for all five implicit rules.

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
- ICMP types and codes
- IGMP types
- Precedence level
- Differentiated Services Code Point (DSCP) value
- TCP packets with the ACK, FIN, PSH, RST, SYN, or URG bit set
- Established TCP connections

MAC ACLs support the following additional filtering options:

- Layer 3 protocol
- VLAN ID
- Class of Service (CoS)

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.

Note The range operator is inclusive of boundary values.

These LOUs minimize the number of ternary content addressable memory (TCAM) entries needed to perform these operations. A maximum of two LOUs are allowed for each feature on an interface. For example an ingress RACL can use two LOUs, and a QoS feature can use two LOUs. If an ACL feature requires more than two arithmetic operations, the first two operations use LOUs, and the remaining access control entries (ACEs) get expanded.

The following guidelines determine when the device stores operator-operand couples in LOUs:

- If the operator or operand differs from other operator-operand couples that are used in other rules, the couple is stored in an LOU.

  For example, the operator-operand couples "gt 10" and "gt 11" would be stored separately in half an LOU each. The couples "gt 10" and "lt 10" would also be stored separately.

- Whether the operator-operand couple is applied to a source port or a destination port in the rule affects LOU usage. Identical couples are stored separately when one of the identical couples is applied to a source port and the other couple is applied to a destination port.

  For example, if a rule applies the operator-operand couple "gt 10" to a source port and another rule applies a "gt 10" couple to a destination port, both couples would also be stored in half an LOU, resulting in the use of one whole LOU. Any additional rules using a "gt 10" couple would not result in further LOU usage.

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, 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.
- IPv6 PACL TCAM is not supported for Cisco NX-OS 3000 Series switches.
- 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
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	

TCAM ACL Region	Default Size	Minimum Size	Incremental Size	Maximum Size
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

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.

VACLs have the following prerequisite:

- Ensure that the IP ACL that you want to use in the VACL exists and is configured to filter traffic in the manner that you need for this application.

Guidelines and Limitations for ACLs

IP ACLs have the following configuration guidelines and limitations:

- You cannot configure the set-vlan option on the tap-aggregation policy. The set-vlan and strip-vlan options are specific to OpenFlow.
- 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 4 tcp-option-lengths in the ACEs, which includes 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 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.
- Only 62 unique ACLs can be configured. Each ACL takes one label. If same ACL is configured on multiple interfaces, the same label is shared; but if each ACL has unique entries, the ACL labels are not shared and that label limit is 62.
- 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.
- To use the **match-local-traffic** option for all inbound and outbound traffic, you must first enable the ACL in the software.
- An RACL applied on a Layer 3 physical or logical interface does not match multicast traffic. If multicast traffic must be blocked, use a PACL instead.
- You cannot configure egress RACLs on L3 port channels.
- IPv4 ACL logging in the egress direction is not supported.

VACLs have the following configuration:

- We recommend that you perform ACL configurations 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.
- ACL statistics are not supported if the DHCP snooping feature is enabled.
- If an IPv4 ACL, applied as a VLAN ACL, contains one or more ACEs with logical operators for TCP/UDP port numbers, the port numbers are matched in the ingress direction but ignored in the egress direction.
- One VLAN access map can match only one IP ACL.
- An IP ACL can have multiple permit/deny ACEs.
- One VLAN can have only one access map applied.

Default ACL Settings

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

Table 3: 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 VACL parameters.

Table 4: Default VACL Parameters

Parameters	Default
VACLs	No IP 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 switch and add rules to it.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# {**ip** | **ipv6**} **access-list** *name*	Creates the IP ACL and enters IP ACL configuration mode. The *name* argument can be up to 64 characters.
Step 3	switch(config-acl)# [*sequence-number*] {**permit** | **deny**} *protocol source destination*	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.
Step 4	(Optional) switch(config-acl)# **statistics**	Specifies that the switch maintains global statistics for packets that match the rules in the ACL.
Step 5	(Optional) switch# **show** {**ip** | **ipv6**} **access-lists** *name*	Displays the IP ACL configuration.

	Command or Action	Purpose
Step 6	(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
switch(config-acl)# statistics
```

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.

	Command or Action	Purpose

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:
	`switch(config-if)# exit`
	`switch(config)#`
		Updates the configuration and exits interface configuration mode.

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	**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.
Step 12	**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).

	Command or Action	Purpose
Step 13	(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. No other information of active flows will be displayed specifically all the unsupported options.

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**} **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(config-acl)# [**no**] **statistics**	Specifies that the switch maintains global statistics for packets that match the rules in the ACL.

	Command or Action	Purpose
		The **no** option stops the switch from maintaining global statistics for the ACL.
Step 7	(Optional) switch#**show ip access-lists** *name*	Displays the IP ACL configuration.
Step 8	(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 13

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.

	Command or Action	Purpose
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	**interface mgmt** *port*
	Example:
	`switch(config)# interface mgmt0`
	`switch(config-if)#`
		Enters configuration mode for the management interface.

Step 3	**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 4	(Optional) **show running-config aclmgr** **Example:** `switch(config-if)# show running-config aclmgr`	Displays the ACL configuration.
Step 5	(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.
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 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.

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.

	Command or Action	Purpose
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 tunnel** *tunnel-number* • switch(config)# **interface vlan** *vlan-ID* • 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** | **out**} • switch(config-if)# **ipv6 traffic-filter** *access-list* {**in** | **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	(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.

Verifying the IP ACL Configuration

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

Command	Purpose
show hardware access-list tcam region	Displays the TCAM sizes that will be applicable on the next reload of the device.
show ip access-lists	Displays the IPv4 ACL configuration.
show ipv6 access-lists	Displays the IPv6 ACL configuration.
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. No other information of active flows will be displayed specifically all the unsupported options.

Command	Purpose
show logging ip access-list status	Displays the deny maximum flow count, the current effective log interval, and the current effective threshold value.
show running-config acllog	Displays the ACL log running configuration.
show running-config aclmgr [**all**]	Displays the ACL running configuration, including the IP ACL configuration and the interfaces to which IP ACLs are applied. **Note** This command displays the user-configured ACLs in the running configuration. The **all** option displays both the default (CoPP-configured) and user-configured ACLs in the running configuration.
show startup-config acllog	Displays the ACL log startup configuration.
show startup-config aclmgr [**all**]	Displays the ACL startup configuration. **Note** This command displays the user-configured ACLs in the startup configuration. The **all** option displays both the default (CoPP-configured) and user-configured ACLs in the startup configuration.

Monitoring and Clearing IP ACL Statistics

Use the **show ip access-lists** or **show ipv6 access-list** command to display statistics about an IP ACL, including the number of packets that have matched each rule. For detailed information about the fields in the output from this command, see the *Command Reference* for your Cisco Nexus device.

Note The mac access-list is applicable to non-IPv4 and non-IPv6 traffic only.

Procedure

- switch# **show** {**ip** | **ipv6**} **access-lists** *name*

  Displays IP ACL configuration. If the IP ACL includes the **statistics** command, then the **show ip access-lists** and **show ipv6 access-list** command output includes the number of packets that have matched each rule.

- switch#**show ip access-lists** *name*

  Displays IP ACL configuration. If the IP ACL includes the **statistics** command, then the **show ip access-lists** command output includes the number of packets that have matched each rule.

- switch# **clear** {**ip** | **ipv6**} **access-list counters** [*access-list-name*]

  Clears statistics for all IP ACLs or for a specific IP ACL.

- switch# **clear ip access-list counters** [*access-list-name*]

  Clears statistics for all IP ACLs or for a specific IP ACL.

Triggering the RACL Consistency Checker

You can manually trigger the RACL consistency checker to compare the hardware and software configuration of the ingress and egress RACLs of a module and display the results. To manually trigger the RACL consistency checker and display the results, use the following command in any mode:

Procedure

	Command or Action	Purpose
Step 1	**show consistency-checker racl module** *slot*	Starts an RACL consistency check on the specified module and displays the results.

Example

This example shows how to trigger an RACL consistency check and display the results:

```
switch# show consistency-checker racl module 1
Validates RACL on up interfaces:
Consistency Check: FAILED

Found consistencies for following Interfaces:
   Ethernet1/9 (in)
   Ethernet1/9 (out)
   Ethernet1/17 (in)
   Ethernet1/17 (out)

Found inconsistencies for following Interfaces and EID:
   Ethernet1/3 (in)
   Ethernet1/3 (out)
```

Configuring ACL Using HTTP Methods to Redirect Requests

Note 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 4 tcp-option-lengths in the ACEs, which includes 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.

The following HTTP methods can be redirected:

- connect
- delete
- get
- head
- post
- put
- trace

Configure the ACL CLI to redirect specific HTTP methods to a server.

Before you begin

- Create an IP access list.
- Enable the double wide TCAM for the IFACL region using the CLI **hardware profile tcam region ifacl 512 double-wide** command . This command applies to the global configuration and only on Trident2 based Cisco Nexus 3000 Series switches. Reload the switch for this configuration to take into effect.
- Enable tap-aggregation feature to redirect the packets to another interface using the CLI **hardware profile tap-aggregation** command. This command applies to global configuration. Reload the switch for this configuration to take into effect.

Procedure

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

Step 2	switch(config-acl)# **permit** *protocol source* any http-method ?
	Example:
	`switch(config-acl)# permit tcp 1.1.1.1/32 any http-method ?`
	`connect  Match http packets with`
		Configures the ACL CLI to redirect specific HTTP methods to a server.

	Command or Action	Purpose

	CONNECT method [0x434f4e4e]
	  delete Match http packets with DELETE
	 method [0x44454c45]
	  get Match http packets with GET
	method [0x47455420]
	  head Match http packets with HEAD
	 method [0x48454144]
	  post Match http packets with POST
	 method [0x504f5354]
	  put Match http packets with PUT
	method [0x50555420]
	  trace Match http packets with TRACE
	 method [0x54524143]	

Step 3	(Optional) switch# **show ip access-lists** *name*	Displays the IP ACL configuration.
Step 4	(Optional) switch# **show run interface** <*x/y*>	Displays the interface configuration.

Example

In the following example, an Ethernet interface 1/33 is receiving HTTP traffic. Ethernet interface 1/34 is the egress interface. Enable mode **tap-aggregation** on the egress interface. Create an ACL to match the traffic. Configure the redirect HTTP get method that matches the ACL to Ethernet interface 1/34. Apply the ACL to the port where the HTTP traffic is received. Any HTTP get traffic that hits the ACL on Ethernet 1/33 is redirected to the destination interface, for example, Ethernet 1/34. The same steps can be used for the other listed methods.

Troubleshooting Information—In case the ACL is not hit or the packets are not redirected, ensure that double wide TCAM is enabled. Ensure that tap aggregation is enabled. Ensure both source and destination ports are in STP forwarding state in the same VLAN. Ensure that the ACL is programmed in TCAM using the **sh platform afm info attachment interface <interface>** command. The HTTP redirect feature does not work on Layer 3 ports.

```
switch# configure terminal
switch(config)# interface Ethernet 1/33

L3-QI2-CR-one(config)## interface Ethernet 1/34
L3-QI2-CR-one(config-if)# mode tap-aggregation
switch(config)# ip access-list  http-redirect-acl
switch(config-acl)# 10 permit tcp 10.1.1.1/32 10.2.2.2/32 http-method get redirect e1/34
switch(config-acl)# 10 permit tcp any any http-method get tcp-option-length 8 redirect e1/34
switch(config-acl)# 20 permit tcp any any http-method post redirect e1/34
switch(config-acl)# statistics per-entry

switch(config)# interface Ethernet 1/33
switch(config-if)# ip port access-group http-redirect-acl in

switch(config)# show ip access-lists
switch(config)# show run int 1/34
switch(config)# show hardware access-list interface 1/34
```

Information About VLAN ACLs

A VLAN ACL (VACL) is one application of an IP ACL. You can configure VACLs to apply to all packets that are bridged within a VLAN. VACLs are used strictly for security packet filtering. VACLs are not defined by direction (ingress or egress).

VACLs and Access Maps

VACLs use access maps to link an IP ACL to an action. The switch takes the configured action on packets that are permitted by the VACL.

VACLs and Actions

In access map configuration mode, you use the **action** command to specify one of the following actions:

- Forward—Sends the traffic to the destination determined by normal operation of the switch.
- Drop—Drops the traffic.

Statistics

The Cisco Nexus device can maintain global statistics for each rule in a VACL. If a VACL is applied to multiple VLANs, the maintained rule statistics are the sum of packet matches (hits) on all the interfaces on which that VACL is applied.

Note The Cisco Nexus device does not support interface-level VACL statistics.

For each VLAN access map that you configure, you can specify whether the switch maintains statistics for that VACL. This allows you to turn VACL statistics on or off as needed to monitor traffic filtered by a VACL or to help troubleshoot VLAN access-map configuration.

Configuring VACLs

Creating or Changing a VACL

You can create or change a VACL. Creating a VACL includes creating an access map that associates an IP ACL with an action to be applied to the matching traffic.

Procedure

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

	Command or Action	Purpose
Step 2	switch(config)# **vlan access-map** *map-name*	Enters access map configuration mode for the access map specified.
Step 3	switch(config-access-map)# **match ip address** *ip-access-list*	Specifies an IPv4 and IPv6 ACL for the map.
Step 4	switch(config-access-map)# **action** {**drop** | **forward**}	Specifies the action that the switch applies to traffic that matches the ACL.
Step 5	(Optional) switch(config-access-map)# [**no**] **statistics**	Specifies that the switch maintains global statistics for packets matching the rules in the VACL. The **no** option stops the switch from maintaining global statistics for the VACL.
Step 6	(Optional) switch(config-access-map)# **show running-config**	Displays the ACL configuration.
Step 7	(Optional) switch(config-access-map)# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Removing a VACL

You can remove a VACL, which means that you will delete the VLAN access map.

Be sure that you know whether the VACL is applied to a VLAN. The switch allows you to remove VACLs that are current applied. Removing a VACL does not affect the configuration of VLANs where you have applied the VACL. Instead, the switch considers the removed VACL to be empty.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **no vlan access-map** *map-name*	Removes the VLAN access map configuration for the specified access map.
Step 3	(Optional) switch(config)# **show running-config**	Displays ACL configuration.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Applying a VACL to a VLAN

You can apply a VACL to a VLAN.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# [**no**] **vlan filter** *map-name* **vlan-list** *list*	Applies the VACL to the VLANs by the list that you specified. The **no** option unapplies the VACL. The **vlan-list** command can specify a list of up to 32 VLANs, but multiple **vlan-list** commands can be configured to cover more than 32 VLANs.
Step 3	(Optional) switch(config)# **show running-config**	Displays ACL configuration.
Step 4	(Optional) switch(config)# **copy running-config startup-config**	Copies the running configuration to the startup configuration.

Verifying VACL Configuration

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

Command or Action	Purpose
switch# **show running-config aclmgr**	Displays ACL configuration, including VACL-related configuration.
switch# **show vlan filter**	Displays information about VACLs that are applied to a VLAN.
switch#**show vlan access-map**	Displays information about VLAN access maps.

Displaying and Clearing VACL Statistics

To display or clear VACL statistics, perform one of the following tasks:

Procedure

- switch# **show vlan access-list**

  Displays VACL configuration. If the VLAN access-map includes the **statistics** command, then the **show vlan access-list** command output includes the number of packets that have matched each rule.

- switch# **clear vlan access-list counters**

  Clears statistics for all VACLs or for a specific VACL.

Configuration Examples for VACL

The following example shows how to configure a VACL to forward traffic permitted by an IP ACL named acl-ip-01 and how to apply the VACL to VLANs 50 through 82:

```
switch# configure terminal
switch(config)# vlan access-map acl-ip-map
switch(config-access-map)# match ip address acl-ip-01
switch(config-access-map)# action forward
switch(config-access-map)# exit
 switch(config)# vlan filter acl-ip-map vlan-list 50-82
```

Configuring the LOU Threshold

You can configure the LOU threshold. When the number of expanded ACEs exceeds this threshold, the device stores them in an LOU register. Otherwise, the device stores these ACEs as TCAM entries. This configuration takes effect only for the next ACL configuration. All existing ACL configurations either in TCAM or LOU register are not affected by this configuration. In order for the changes to take effect, you have to use the **copy r s** command and reload the box.

Note The expanded ACEs are not stored if the TCAM or all 24 LOU registers are full.

Procedure

	Command or Action	Purpose
Step 1	switch# **configure terminal**	Enters global configuration mode.
Step 2	switch(config)# **hardware profile tcam lou-threshold** *value*	Configures the LOU threshold and the LOU expansion threshold takes effect for the new policies. It is recommended to save the configuration and reload so that the threshold takes effect on the already existing policies. The threshold values range from 1 to 100, and the default LOU threshold value is 1.

Example

This example shows how to configure the LOU threshold:

```
switch# configure terminal
switch(config)# hardware profile tcam lou-threshold 20
switch(config)# copy running-config startup-config
```

```
switch(config)# reload
LOU expansion threshold changed to 20
```

Configuring ACL TCAM Region Sizes

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

Procedure

	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**} | {**ipv6-span** | **span** } | {**ipv6-span-l2** | **span** } | {**spanv6** |**span** } | {**spanv6-12** |**span**} | **vacl** } {**fhs**} *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. The maximum number of entries is 1500. • **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. For FHS, the range is from 0-4096.

	Command or Action	Purpose
		Note **vacl** and **e-vacl** TCAM regions should be set to the same size.
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 profile tcam region** **Example:** `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** **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 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-config 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:

```
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'
```

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)#
```

```
switch(config)# hardware profile  tcam  region  ipv6-span  512
Warning: Please save config and reload the system for the configuration to take effect
switch(config)# hardware  profile  tcam  region  spanv6 qualify  udf  udf1
[SUCCESS] Changes to UDF qualifier set will be applicable only after reboot.
```

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

```
switch(config)# show hardware profile tcam region
         sup size =   16
        vacl size =  640
       ifacl size =  496
         qos size =  256
       rbacl size =    0
        span size =    0
        racl size = 1536
      e-racl size =  256
      e-vacl size =  640
      qoslbl size =    0
        ipsg size =    0
      arpacl size =    0
   ipv6-racl size =    0
 ipv6-e-racl size =    0
    ipv6-sup size =    0
    ipv6-qos size =    0
```

```
switch(config)# show hardware profile tcam region
         sup size =   16
        vacl size =  640
       ifacl size =  496
         qos size =  256
       rbacl size =    0
        span size =    0
        racl size = 1536
      e-racl size =  256
      e-vacl size =  640
      qoslbl size =    0
      arpacl size =    0
   ipv6-racl size =    0
 ipv6-e-racl size =    0
    ipv6-sup size =    0
    ipv6-qos size =    0
```

Note On Cisco Nexus 3000 Series switches, you must carve the switch RACL TCAM regions in order to make IGMP and PIM work on Layer 3 interfaces. Some system default Multicast ACLs that are installed in the RACL regions are required for IGMP and PIM to work on Layer 3 interfaces.

Note If the config-control property is set to YES in the XML hierarchy definition file, then it is possible for the memory object to use a faulty bit map to report the error.

Reverting to the Default TCAM Region Sizes

Procedure

	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** | **e-racl**} | **ifacl** | **ipsg** | **qos**} |**qoslbl** | **racl**} | **vacl** } *tcam_size*	Reverts the configuration to the default ACL TCAM size.
Step 3	(Optional) **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)# **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.

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#
```