



Configuring Access Control Lists

This chapter contains the following sections:

- [About ACLs, on page 1](#)
- [Configuring IP ACLs, on page 9](#)
- [Information About VLAN ACLs, on page 15](#)
- [Configuring VACLs, on page 16](#)
- [Configuration Examples for VACL, on page 18](#)
- [Configuring ACL TCAM Region Sizes, on page 18](#)
- [Configuring ACLs on Virtual Terminal Lines, on page 21](#)
- [Configuring Wideflow IFACL Redirect on IP Port ACLs, on page 24](#)
- [Configuring Redirect Action, on page 27](#)

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	<p>An ACL is considered a port ACL when you apply it to one of the following:</p> <ul style="list-style-type: none"> • Ethernet interface • Ethernet port-channel interface <p>When a port ACL is applied to a trunk port, the ACL filters traffic on all VLANs on the trunk port.</p>	IPv4 ACLs
Router ACL	<ul style="list-style-type: none"> • VLAN interfaces <p>Note You must enable VLAN interfaces globally before you can configure a VLAN interface.</p> <ul style="list-style-type: none"> • 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
VLAN ACL (VACL)	<p>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.</p>	IPv4 ACLs
VTY ACL	VTYs	IPv4 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

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

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

This section describes some of the options that you can use when you configure a rule.

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 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.

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

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

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.

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 on the Cisco Nexus device, 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.
- The total TCAM depth is 4000 entries shared between ingress and egress, which can be carved in 16 entries blocks.
- TCAM supports 256 statistic entries per ACL feature.
- 64 ACL L4OPs are supported, 32 in each direction.
- 2 L4OPs are supported per label in each direction. Each label can be shared across multiple interfaces for same ACL.
- 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
SUP (ingress)	112	48	16
PACL (ingress)	400	0	16
VACL (ingress), VACL (egress)	640 (ingress), 640 (egress)	0 (ingress), 0 (egress)	16
RACL (ingress)	1536	0	16
QOS (ingress), QOS (egress)	192 (ingress), 64 (egress)	16 (ingress), 64 (egress)	16
E-VACL (egress)	640	0	16
E-RACL (egress)	256	0	16
NAT	256	0	16

Licensing Requirements for ACLs

For a complete explanation of Cisco NX-OS licensing recommendations and how to obtain and apply licenses, see the [Cisco NX-OS Licensing Guide](#).

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:

- 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.

- 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).
- To apply an IP ACL to a VLAN interface, you must have enabled VLAN interfaces globally.
- 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.
- Egress RACLs and VACLs should not be applied in warp mode, and it is not supported.
- Egress ACLs cannot be applied to multicast traffic.
- Egress ACL Logging not supported on Cisco Nexus 3548 platform.
- Although Ingress RACLs on SVI are supported for multicast traffic, if the ACL defining the multicast group where the traffic must be sent to or sourced from includes the **log** keyword, the ingress RACL application on SVI is not supported.
- To match the Ingress RACL ACEs for multicast traffic on SVI, the ACE should include a match on the multicast DIP. Also, before installing these ACEs, you must enable the **RACL-bridging using the hardware profile tcam mcast racl-bridge** command.
- PACL cannot be applied in warp mode.
- The same ingress RACL on an SVI and on a Layer 3 interface cannot share TCAM resources, and they individually use up TCAM resources. However, they share ACL statistics resources. If the RACL TCAM is almost depleted before upgrade, RACL application may fail after upgrade. If this happens, you can carve the RACL TCAM.
- ARP ACLs are not supported on the Nexus 3500 platform.
- Ingress RACL applied to physical or logical Layer 3 interface is supported. For the ingress RACL to be applied to the Layer 3 SVI, you can use the *hardware profile tcam mcast racl-bridge* configuration as a workaround to match multicast traffic.
- Upgrade from Cisco NX-OS Release 7.0(3)I7(6) or below, Cisco NX-OS Release 9.3(1) to 9.3(2) or above with default lou threshold config will set lou threshold as 1.
- In the Cisco Nexus 3548 Series switches, RACL with ACL log option will not take into effect as the sup-redirect ACLs will have higher priority for the traffic destined to SUP.

Below are the guidelines and limitations for wide IFACL:

- Same egress ports on two different flows with different **SET_VLAN id** is not possible if the ingress match VLAN is same for both the flows.
- Wide flow IFACL Redirect Action is Supported only on Trunk Ports.
- No other ACL feature will be supported on flow-redirect ports except PACL. PACL entries (Wide flow or not) will be installed in FIBACL TCAM & not in ACL TCAM like normal PACL when PACL_WIDE TCAM region is carved.
- During port flap, entries will not be removed from TCAM. They will stay as is like other security ACLs.

- Port range match give in CLI will be expanded in value & mask for L4 ports before writing to TCAM and LOU hardware resources will not be used. No impact to user and no impact in terms of existing scale of flows.
- Only Redirect/Set-vlan/Strip-vlan & Drop Actions are Supported. No Support for PUNT Action.
- Log keyword is not supported for wide IFACL ACLs.
- Max 4000 redirect ACLs are supported irrespective of TCAM size.
- Max 4k ACEs with stats can be supported.
- VLAN Range allowed for match & set/strip: 1 – 4094.
- ACE Match on TCP flags are not supported.
- Before changing the TCAM configuration from **ifacl-wide** to **ifacl**, ensure that all wideflow ACL's configurations are removed from interfaces.
- If ingress packets have with same VLAN matching wideflow ACEs with and without VLAN match conditions along with strip_vlan, VLAN header gets stripped even for packets matching no strip-vlan ACE.

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 .
Object groups	No object groups exist by default.

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 ACL on the switch and add rules to it.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# ip access-list <i>name</i>	Creates the IP ACL and enters IP ACL configuration mode. The <i>name</i> argument can be up to 64 characters.
Step 3	switch(config-acl)# [<i>sequence-number</i>] { permit deny } <i>protocol source destination</i>	Creates a rule in the IP ACL. You can create many rules. The <i>sequence-number</i> 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 <i>Command Reference</i> for the specific Cisco Nexus device.
Step 4	(Optional) switch(config-acl)# statistics	Specifies that the switch maintains global statistics for packets that matches the rules in the ACL.
Step 5	(Optional) switch# show ip access-lists <i>name</i>	Displays the IP ACL configuration.
Step 6	(Optional) switch# copy running-config startup-config	Copies the running configuration to the startup configuration.

Example

The following 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
```

Changing an IP ACL

You can add and remove rules in an existing IPv4 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 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)# [<i>sequence-number</i>] permit deny } <i>protocol source destination</i>	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 <i>sequence-number</i> 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 <i>Command Reference</i> for your Cisco Nexus device.
Step 5	(Optional) switch(config-acl)# no { <i>sequence-number</i> permit deny } <i>protocol source destination</i> }	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 <i>Command Reference</i> 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. 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 11

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 access-list <i>name</i>	Removes the IP ACL that you specified by name from the running configuration.
Step 3	switch(config)# no ip access-list <i>name</i>	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	switch(config)# resequence ip access-list <i>name</i> <i>starting-sequence-number</i> <i>increment</i>	Assigns sequence numbers to the rules contained in the ACL, where the first rule receives the starting sequence number that you specify. Each subsequent rule receives a number larger than the preceding rule. The difference in numbers is determined by the increment that you specify. The <i>starting-sequence-number</i> argument and the <i>increment</i> argument can be a whole number between 1 and 4294967295.
Step 3	(Optional) switch# show ip access-lists <i>name</i>	Displays the IP ACL configuration.
Step 4	(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 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 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 [<i>chassis</i> /] <i>slot</i> / <i>port</i> port-channel <i>channel-number</i> }	Enters interface configuration mode for the specified interface.
Step 3	switch(config-if)# ip port access-group <i>access-list in</i>	Applies an IPv4 ACL to the interface or PortChannel. Only inbound filtering is supported with port ACLs. You can apply one port ACL to an interface.
Step 4	(Optional) switch# show running-config	Displays the ACL configuration.
Step 5	(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 ACL to any of the following types of interfaces:

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

ACLs applied to these interface types are considered router ACLs.



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

Before you begin

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

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: <ul style="list-style-type: none"> • switch(config)# interface ethernet <i>slot/port</i> [. <i>number</i>] • switch(config)# interface port-channel <i>channel-number</i> [. <i>number</i>] • switch(config)# interface tunnel <i>tunnel-number</i> • switch(config)# interface vlan <i>vlan-ID</i> • switch(config)# interface mgmt <i>port</i> 	Enters configuration mode for the interface type that you specified.
Step 3	switch(config-if)# ip access-group <i>access-list</i> { in out }	Applies an IPv4 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 IP ACL Configurations

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

Procedure

- switch# **show running-config**

Displays ACL configuration, including IP ACL configuration and interfaces that IP ACLs are applied to.

- switch# **show running-config interface**

Displays the configuration of an interface to which you have applied an ACL.

- switch# **show running-config aclmgr**

Displays ACL configurations and the interfaces the ACLs are applied to.

Example

For detailed information about the fields in the output from these commands, refer to the *Command Reference* for your Cisco Nexus device.

Monitoring and Clearing IP ACL Statistics

Use the **show ip access-lists** 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 traffic only.

Procedure

- 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#**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 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.

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.
Step 2	switch(config)# vlan access-map <i>map-name</i>	Enters access map configuration mode for the access map specified.
Step 3	switch(config-access-map)# match ip address <i>ip-access-list</i>	Specifies an IPv4 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 <i>map-name</i>	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 <i>map-name</i> vlan-list <i>list</i>	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:

Procedure

- 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 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: <pre>switch# configure terminal switch(config)#</pre>	Enters global configuration mode.
Step 2	hardware profile tcam region {arpacl e-racl} ifacl nat qos} qoslbl racl} vacl } tcam_size	Changes the ACL TCAM region size. <ul style="list-style-type: none"> • arpacl—Configures the size of the Address Resolution Protocol (ARP) ACL (ARPA) TCAM region.

	Command or Action	Purpose
		<ul style="list-style-type: none"> • 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. • nat—Configures the size of the NAT 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. <p>Note vacl and e-vacl TCAM regions should be set to the same size.</p>
Step 3	<p>copy running-config startup-config</p> <p>Example:</p> <pre>switch(config)# copy running-config startup-config</pre>	<p>Saves the change persistently through reboots and restarts by copying the running configuration to the startup configuration.</p>
Step 4	<p>switch(config)# show hardware profile tcam region</p> <p>Example:</p> <pre>switch(config)# show hardware profile tcam region</pre>	<p>Displays the TCAM sizes that will be applicable on the next reload of the switch.</p>
Step 5	<p>switch(config)# reload</p> <p>Example:</p> <pre>switch(config)# reload</pre>	<p>Copies the running configuration to the startup configuration.</p> <p>Note The new size values are effective only upon the next reload after saving the copy running-config to startup-config.</p>

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 how to configure the TCAM VLAN ACLs on a switch:

```
switch# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
switch(config)# hardware profile tcam region vacl 512
[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 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
```

This example shows how to determine TCAM utilization for particular region. There are 5 RACL entries in this example:

```
switch(config)# show system internal aclqos platform mtc info tcam 0 region racl
  racl TCAM configuration for ASIC id 0:
[   sup tcam]: range 0 - 47
[   vacl tcam]: range 512 - 1087
[   ifacl tcam]: range 112 - 511
[   qos tcam]: range 3712 - 3903
[   rbacl tcam]: range 0 - 0
[   span tcam]: range 0 - 0
[   racl tcam]: range 1984 - 3455 *
[   e-racl tcam]: range 3456 - 3711
[   e-vacl tcam]: range 1088 - 1727
[   qoslbl tcam]: range 0 - 0
[   ipsg tcam]: range 0 - 0
[   arpacl tcam]: range 0 - 0
[  ipv6-racl tcam]: range 0 - 0
[ipv6-e-racl tcam]: range 0 - 0
[  ipv6-sup tcam]: range 0 - 0
[  ipv6-qos tcam]: range 0 - 0
[   nat tcam]: range 1728 - 1983
```

```

[ e-qos tcam]: range 3904 - 3967
[ pbr tcam]: range 0 - 0
[ ipv6-pbr tcam]: range 0 - 0
[ copp tcam]: range 48 - 111

TCAM [racl tcam]: [v:1, size:1472, start:1984 end:3455]
In use tcam entries: 5
3451-3455
Link Local Entries:
nat size = 256

```

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 nat 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 RAACL 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-config 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 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:	Copies the running configuration to the startup configuration.

	Command or Action	Purpose
	switch# copy running-config startup-config	

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#
```

Configuring Wideflow IFACL Redirect on IP Port ACLs

Until Cisco NX-OS Release 10.3(2)F the Tap Aggregation functionality in Cisco Nexus 3548 series switches are supported using Openflow. For more information, see [Configuring the Cisco OpenFlow Agent](#).

Beginning with Cisco NX-OS Release 10.3(3)F Openflow is not supported on Cisco Nexus 3548 series switches. To cater all openflow or Tap aggregation functionalities, ACL redirect with wideflow feature is introduced with additional new match command options (srcmac, dstmac & vlan) and new actions (setvlan, strip-vlan).

Beginning with Cisco NX-OS Release 10.3(3)F, new CLI options are added in conjunction with keyword **wideflow** in the existing IP ACL CLI. Keyword **wideflow** protects the new CLI options and it is only enabled for Cisco Nexus 3548 switches.

Before you begin

To enable wideflow new command options, **IFACL-WIDE TCAM** needs to be configured. This requires a copy running configuration to the startup configuration and reload of the device. Hardware profile forwarding-mode will change from normal to flow-redirect post reload. For more information, see [Achieving OpenFlow Functionality](#).



- Note**
- While changing from IFACL to IFACL-WIDE TCAM, make sure that all the existing IP access-lists are removed from interfaces and global configurations.
 - After changing to IFACL-WIDE TCAM, legacy ACL's cannot be applied under interfaces.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# ip access-list name	Creates the IP ACL and enters IP ACL configuration mode. The <i>name</i> argument can be up to 64 characters.
Step 3	switch(config-acl)# [<i>sequence-number</i>] { permit deny } <i>protocol source destination redirect redirect-ports wideflow</i>	Following are the options under wideflow. <ul style="list-style-type: none"> • dstmac—Configure the destination MAC Address. • srcmac—Configure the source MAC Address. • vlan—Configure the Vlan number.
Step 4	switch(config-acl)# [<i>sequence-number</i>] { permit deny } <i>protocol source destination redirect redirect-ports wideflow dstmac destination MAC address</i>	Following are the options under dstmac . <ul style="list-style-type: none"> • E.E.E—Destination wildcard bits (Option 1). • EE-EE-EE-EE-EE-EE—Destination wildcard bits (Option 2). • EE:EE:EE:EE:EE:EE—Destination wildcard bits (Option 3). • EEEE.EEEE.EEEE—Destination wildcard bits (Option 4).
Step 5	switch(config-acl)# [<i>sequence-number</i>] { permit deny } <i>protocol source destination redirect redirect-ports wideflow srcmac source MAC address</i>	Following are the options under srcmac . <ul style="list-style-type: none"> • E.E.E—Source MAC address (Option 1). • EE-EE-EE-EE-EE-EE—Source MAC address (Option 2). • EE:EE:EE:EE:EE:EE—Source MAC address (Option 3). • EEEE.EEEE.EEEE—Source MAC address (Option 4). • any—Any source address.

	Command or Action	Purpose
Step 6	switch(config-acl)# [<i>sequence-number</i>] permit deny } <i>protocol source destination redirect redirect-ports wideflow vlan</i>	Enter the Vlan number range from 0 to 4095.

Example

Following is the configuration example:

Step 1: If the switch is in openflow forwarding-mode, perform the following steps:



Note If the switch is in normal forwarding-mode, skip Step 1 and go to Step 2 directly.

- Remove all openflow configurations.
- Change hardware profile forwarding-mode to normal.
- Copy the running configuration to the startup configuration.
- Reload the switch.

```
switch#configure terminal
switch(config)# no openflow
switch(config)# no feature openflow
switch(config)# [optional] no hardware profile openflow forward-pdu
switch(config)# hardware profile forwarding-mode normal
switch(config)# copy r s
switch(config)# reload
```

Step 2: Upgrade to Cisco Nexus Release 10.3(3)F or later releases.

Step 3: After the switch is booted in Cisco Nexus Release 10.3(3)F or later releases, configure **TCAM for IFACL-WIDE** as follows:

```
switch# configure terminal
switch(config)# hardware profile tcam region ifacl 0
switch(config)# hardware profile tcam region ifacl-wide 4096
switch(config)# copy r s
switch(config)# reload
switch(config)# [optional] hardware profile flow-redirect forward-pdu
```

Following is the example for IP access-list configurations using redirect and wideflow commands:

```
switch# configure terminal
switch(config)# ip access-list ACL
switch(config-acl)# 10 permit ip host 1.1.1.1 host 1.1.1.2 dscp 52 redirect
Ethernet1/2,portchannel1 strip-vlan wideflow srcmac 00:16:3e:33:e1:84 0.0.0 dstmac
00:16:3e:4d:d6:dd 0.0.0 vlan 1000
switch(config-acl)# 20 permit icmp host 2.2.2.1 host 2.2.2.2 redirect
Ethernet1/34,portchannel2 wideflow
switch(config-acl)# 30 permit tcp host 3.3.3.1 host 3.3.3.2 dscp 28 redirect
Ethernet1/2,port-channel1 set-vlan 1002 wideflow srcmac 00:16:3e:12:e9:c4 0.0.0 dstmac
00:16:3e:0f:6a:48 0.0.0 vlan 1001
switch(config-acl)# 40 permit udp host 4.4.4.1 host 4.4.4.2 precedence 7 redirect
Ethernet1/2,port-channel1 wideflow srcmac 00:16:3e:07:aa:53 0.0.0 dstmac 00:16:3e:79:e4:a8
0.0.0 vlan 1000
switch(config-acl)# 50 permit ethertype 0x0806 redirect Ethernet1/48 wideflow
```

Following is the example for applying IP ACL with redirect and wideflow commands under interface:

```
switch# configure terminal
switch(config)# interface Ethernet1/1
switch(config-if)# mode flow-redirect
switch(config-if)# ip port access-group ACL in
switch(config-if)# end
```

Configuring Redirect Action

Redirect Action in CLI syntax must be present before **wideflow** keyword. Redirect Action Configuration is not accepted if **wideflow** keyword is missing. This check is performed at run time once user has entered the command.

Procedure

	Command or Action	Purpose
Step 1	switch# configure terminal	Enters configuration mode.
Step 2	switch(config)# ip access-list name	Creates the IP ACL and enters IP ACL configuration mode. The <i>name</i> argument can be up to 64 characters.
Step 3	switch(config-acl)# [<i>sequence-number</i>] { permit deny } <i>protocol source destination</i> redirect <i>redirect</i>	Following are the options under redirect . <ul style="list-style-type: none"> • redirect—Redirect to interface(s). Syntax example: <code>redirect Ethernet1/1,Ethernet1/2,port-channel1</code>. • wideflow—Wide-flow options (mandatory).
Step 4	switch(config-acl)# [<i>sequence-number</i>] { permit deny } <i>protocol source destination</i> redirect <i>redirect</i>	Following are the optional commands under redirect redirect . <ul style="list-style-type: none"> • redirect—Redirect to interface(s). Syntax example: <code>redirect Ethernet1/1,Ethernet1/2,port-channel1</code>. • set-vlan—Set vlan value for traffic egressing via redirect ports(s). • strip-vlan—Send vlan untagged packet from redirect port(s). • wideflow—Wide-flow options (mandatory).

