



Security Configuration Guide: Access Control Lists, Cisco IOS XE Release 3SE (Cisco WLC 5700 Series)

Americas Headquarters

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
<http://www.cisco.com>
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: <http://www.cisco.com/go/trademarks>. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

© 2015 Cisco Systems, Inc. All rights reserved.



CONTENTS

CHAPTER 1

Creating an IP Access List to Filter TCP Flags 1

- Finding Feature Information 1
- Prerequisites for Creating an IP Access List to Filter TCP Flags 2
- Information About Creating an IP Access List to Filter TCP Flags 2
 - Benefits of Filtering on TCP Flags 2
 - TCP Flags 2
- How to Create an IP Access List to Filter TCP Flags 3
 - Filtering Packets That Contain TCP Flags 3
- Configuration Examples for Configuring an IP Access List to Filter TCP Flags 5
 - Example: Filtering Packets That Contain TCP Flags 5
- Additional References for Creating an IP Access List to Filter TCP Flags 6
- Feature Information for Creating an IP Access List to Filter TCP Flags 7

CHAPTER 2

IP Named Access Control Lists 9

- Finding Feature Information 9
- Information About IP Named Access Control Lists 10
 - Definition of an Access List 10
 - Named or Numbered Access Lists 10
 - Benefits of IP Access Lists 11
 - Access List Rules 11
 - Helpful Hints for Creating IP Access Lists 12
 - Where to Apply an Access List 13
- How to Configure IP Named Access Control Lists 14
 - Creating an IP Named Access List 14
 - Applying an Access List to an Interface 16
- Configuration Examples for IP Named Access Control Lists 17
 - Example: Creating an IP Named Access Control List 17
 - Example: Applying the Access List to an Interface 17

Additional References for IP Named Access Control Lists 17

Feature Information for IP Named Access Control Lists 18

CHAPTER 3**IP Access List Entry Sequence Numbering 19**

Finding Feature Information 19

Restrictions for IP Access List Entry Sequence Numbering 19

Information About IP Access List Entry Sequence Numbering 20

 Purpose of IP Access Lists 20

 How an IP Access List Works 20

 IP Access List Process and Rules 20

 Helpful Hints for Creating IP Access Lists 21

 Source and Destination Addresses 22

 Wildcard Mask and Implicit Wildcard Mask 22

 Transport Layer Information 22

 Benefits IP Access List Entry Sequence Numbering 23

 Sequence Numbering Behavior 23

How to Use Sequence Numbers in an IP Access List 24

 Sequencing Access-List Entries and Revising the Access List 24

Configuration Examples for IP Access List Entry Sequence Numbering 28

 Example: Resequencing Entries in an Access List 28

 Example: Adding Entries with Sequence Numbers 29

 Example: Entry Without Sequence Number 29

Additional References 30

Feature Information for IP Access List Entry Sequence Numbering 30

CHAPTER 4**Standard IP Access List Logging 33**

Finding Feature Information 33

Restrictions for Standard IP Access List Logging 33

Information About Standard IP Access List Logging 34

 Standard IP Access List Logging 34

How to Configure Standard IP Access List Logging 34

 Creating a Standard IP Access List Using Numbers 34

 Creating a Standard IP Access List Using Names 35

Configuration Examples for Standard IP Access List Logging 37

 Example: Creating a Standard IP Access List Using Numbers 37

Example: Creating a Standard IP Access List Using Names	37
Example: Limiting Debug Output	37
Additional References for Standard IP Access List Logging	37
Feature Information for Standard IP Access List Logging	38

CHAPTER 5**IPv6 PACL Support 39**

Finding Feature Information	39
Prerequisites for IPv6 PACL Support	39
Information About IPv6 PACL Support	40
IPv6 Port-Based Access Control List Support	40
How to Configure IPv6 PACL Support	40
Configuring PACL Mode and Applying IPv6 PACL on an Interface	40
Configuration Examples for IPv6 PACL Support	41
Example: Configuring PACL Mode and Applying IPv6 PACL on an Interface	41
Additional References for IPv6 PACL Support	41
Feature Information for IPv6 PACL Support	42

CHAPTER 6**IPv6 Services—Standard Access Control Lists 45**

Finding Feature Information	45
Information About IPv6 Services--Standard Access Control Lists	45
Access Control Lists for IPv6 Traffic Filtering	45
IPv6 Packet Inspection	46
Access Class Filtering in IPv6	46
How to Configure IPv6 Services--Standard Access Control Lists	46
Configuring IPv6 Services—Standard Access Control Lists	46
Configuration Examples for IPv6 Services--Standard Access Control Lists	49
Example: Configuring IPv6 Services—Standard Access Control Lists	49
Example: Creating and Applying an IPv6 ACL	49
Additional References for IPv6 Services—Standard Access Control Lists	49
Feature Information for IPv6 Services—Standard Access Control Lists	50



CHAPTER 1

Creating an IP Access List to Filter TCP Flags

This module documents the ACL TCP Flags Filtering feature and describes how to use an IP access list to filter IP packets that contain TCP flags. The ACL TCP Flags Filtering feature allows you to select any combination of flags on which to filter. The ability to match on a flag set and on a flag not set gives you a greater degree of control for filtering on TCP flags, thus enhancing security.

The ACL TCP Flags Filtering feature provides a flexible mechanism for filtering on TCP flags. Before this feature, an incoming packet was matched if any TCP flag in the packet matched a flag specified in the access control entry (ACE). This behavior allowed for a security loophole, because packets with all flags set could get past the access control list (ACL).

- [Finding Feature Information, page 1](#)
- [Prerequisites for Creating an IP Access List to Filter TCP Flags, page 2](#)
- [Information About Creating an IP Access List to Filter TCP Flags, page 2](#)
- [How to Create an IP Access List to Filter TCP Flags, page 3](#)
- [Configuration Examples for Configuring an IP Access List to Filter TCP Flags, page 5](#)
- [Additional References for Creating an IP Access List to Filter TCP Flags, page 6](#)
- [Feature Information for Creating an IP Access List to Filter TCP Flags, page 7](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Creating an IP Access List to Filter TCP Flags

Before you perform any of the tasks in this module, you should be familiar with the information in the following modules:

- “IP Access List Overview”
- “Creating an IP Access List and Applying It to an Interface”

Information About Creating an IP Access List to Filter TCP Flags

Benefits of Filtering on TCP Flags

The ACL TCP Flags Filtering feature provides a flexible mechanism for filtering on TCP flags. Previously, an incoming packet was matched as long as any TCP flag in the packet matched a flag specified in the access control entry (ACE). This behavior allows for a security loophole, because packets with all flags set could get past the access control list (ACL). The ACL TCP Flags Filtering feature allows you to select any combination of flags on which to filter. The ability to match on a flag set and on a flag not set gives you a greater degree of control for filtering on TCP flags, thus enhancing security.

Because TCP packets can be sent as false synchronization packets that can be accepted by a listening port, it is recommended that administrators of firewall devices set up some filtering rules to drop false TCP packets.

The ACEs that make up an access list can be configured to detect and drop unauthorized TCP packets by allowing only the packets that have a very specific group of TCP flags set or not set. The ACL TCP Flags Filtering feature provides a greater degree of packet-filtering control in the following ways:

- You can select any desired combination of TCP flags on which to filter TCP packets.
- You can configure ACEs to allow matching on a flag that is set, as well as on a flag that is not set.

TCP Flags

The table below lists the TCP flags, which are further described in RFC 793, *Transmission Control Protocol*.

Table 1: TCP Flags

TCP Flag	Purpose
ACK	Acknowledge flag—Indicates that the acknowledgment field of a segment specifies the next sequence number the sender of this segment is expecting to receive.
FIN	Finish flag—Used to clear connections.
PSH	Push flag—Indicates the data in the call should be immediately pushed through to the receiving user.

TCP Flag	Purpose
RST	Reset flag—Indicates that the receiver should delete the connection without further interaction.
SYN	Synchronize flag—Used to establish connections.
URG	Urgent flag—Indicates that the urgent field is meaningful and must be added to the segment sequence number.

How to Create an IP Access List to Filter TCP Flags

Filtering Packets That Contain TCP Flags

This task configures an access list to filter packets that contain TCP flags and verifies that the access list has been configured correctly.



Note

- TCP flag filtering can be used only with named, extended ACLs.
- The ACL TCP Flags Filtering feature is supported only for Cisco ACLs.
- Previously, the following command-line interface (CLI) format could be used to configure a TCP flag-checking mechanism:

permit tcp any any rst The following format that represents the same ACE can now be used: **permit tcp any any match-any +rst** Both the CLI formats are accepted; however, if the new keywords **match-all** or **match-any** are chosen, they must be followed by the new flags that are prefixed with “+” or “-”. It is advisable to use only the old format or the new format in a single ACL. You cannot mix and match the old and new CLI formats.



Caution

If a device having ACEs with the new syntax format is reloaded with a previous version of the Cisco software that does not support the ACL TCP Flags Filtering feature, the ACEs will not be applied, leading to possible security loopholes.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *access-list-name*
4. [*sequence-number*] **permit tcp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**established**]{**match-any** | **match-all**} {+ | -} *flag-name* [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
5. [*sequence-number*] **deny tcp** *source source-wildcard* [*operator* [*port*]] *destination destination-wildcard* [*operator* [*port*]] [**established**]{**match-any** | **match-all**} {+ | -} *flag-name* [**precedence** *precedence*] [**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
6. Repeat Step 4 or Step 5 as necessary, adding statements by sequence number where you planned. Use the **no** *sequence-number* command to delete an entry.
7. **end**
8. **show ip access-lists** *access-list-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list extended <i>access-list-name</i> Example: Device(config)# ip access-list extended kmd1	Specifies the IP access list by name and enters named access list configuration mode.
Step 4	[<i>sequence-number</i>] permit tcp <i>source source-wildcard</i> [<i>operator</i> [<i>port</i>]] <i>destination destination-wildcard</i> [<i>operator</i> [<i>port</i>]] [established]{ match-any match-all } {+ -} <i>flag-name</i> [precedence <i>precedence</i>] [tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] Example: Device(config-ext-nacl)# permit tcp any any match-any +rst	Specifies a permit statement in named IP access list mode. <ul style="list-style-type: none"> • This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • Use the TCP command syntax of the permit command. • Any packet with the RST TCP header flag set will be matched and allowed to pass the named access list kmd1 in Step 3.

	Command or Action	Purpose
Step 5	<p>[<i>sequence-number</i>] deny tcp <i>source source-wildcard</i> [<i>operator</i> [<i>port</i>]] <i>destination destination-wildcard</i> [<i>operator</i> [<i>port</i>]] [established {match-any match-all} {+ -} <i>flag-name</i>] [precedence <i>precedence</i>] [tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments]</p> <p>Example:</p> <pre>Device(config-ext-nacl)# deny tcp any any match-all -ack -fin</pre>	<p>(Optional) Specifies a deny statement in named IP access list mode.</p> <ul style="list-style-type: none"> • This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • Use the TCP command syntax of the deny command. • Any packet that does not have the ACK flag set, and also does not have the FIN flag set, will not be allowed to pass the named access list <code>kmdl</code> in Step 3. • See the deny(IP) command for additional command syntax to permit upper-layer protocols (ICMP, IGMP, TCP, and UDP).
Step 6	Repeat Step 4 or Step 5 as necessary, adding statements by sequence number where you planned. Use the no <i>sequence-number</i> command to delete an entry.	Allows you to revise the access list.
Step 7	<p>end</p> <p>Example:</p> <pre>Device(config-ext-nacl)# end</pre>	(Optional) Exits the configuration mode and returns to privileged EXEC mode.
Step 8	<p>show ip access-lists <i>access-list-name</i></p> <p>Example:</p> <pre>Device# show ip access-lists kmdl</pre>	<p>(Optional) Displays the contents of the IP access list.</p> <ul style="list-style-type: none"> • Review the output to confirm that the access list includes the new entry.

Configuration Examples for Configuring an IP Access List to Filter TCP Flags

Example: Filtering Packets That Contain TCP Flags

The following access list allows TCP packets only if the TCP flags ACK and SYN are set and the FIN flag is not set:

```
ip access-list extended aaa
 permit tcp any any match-all +ack +syn -fin
end
```

The **show access-list** command has been entered to display the ACL:

```
Device# show access-list aaa
Extended IP access list aaa
 10 permit tcp any any match-all +ack +syn -fin
```

Additional References for Creating an IP Access List to Filter TCP Flags

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Security Commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
Order of access list entries	"Refining an IP Access List"
Access list entries based on time of day or week	"Refining an IP Access List"
Packets with noninitial fragments	"Refining an IP Access List"
Filtering on IP Options, TCP flags, noncontiguous ports, or TTL values	"Creating an IP Access List to Filter IP Options, TCP Flags, Noncontiguous Ports, or TTL Values"
Access to virtual terminal lines	"Controlling Access to a Virtual Terminal Line"
Routing updates and policy routing	"Configuring Routing Protocol-Independent Features" modules in the <i>Cisco IOS IP Routing Protocols Configuration Guide</i>
Traffic identification or classification for features such as congestion avoidance, congestion management, and priority queuing	"Regulating Packet Flow on a Per-Interface Basis--Using Generic Traffic Shaping" module in the <i>Quality of Service Solutions Configuration Guide</i>

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Creating an IP Access List to Filter TCP Flags

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [http://www.cisco.com/go/featurenavigator](#). An account on Cisco.com is not required.

Table 2: Feature Information for Creating an IP Access List to Filter TCP Flags

Feature Name	Releases	Feature Configuration Information
ACL TCP Flags Filtering	Cisco IOS XE Release 3.3SE	<p>This feature provides a flexible mechanism for filtering on TCP flags.</p> <p>The ACL TCP Flags Filtering feature allows you to select any combination of flags on which to filter. The ability to match on a flag set and on a flag not set gives you a greater degree of control for filtering on TCP flags, thus enhancing security.</p> <p>In Cisco IOS XE Release 3.3SE, this feature is supported on Cisco Catalyst 3850 Series Switches and Cisco 5700 Wireless LAN Controllers.</p>



IP Named Access Control Lists

Access control lists (ACLs) perform packet filtering to control the movement of packets through a network. Packet filtering provides security by limiting the access of traffic into a network, restricting user and device access to a network, and preventing traffic from leaving a network. IP access lists reduce the chance of spoofing and denial-of-service attacks, and allow dynamic, temporary user-access through a firewall.

The IP Named Access Control Lists feature gives network administrators the option of using names to identify their access lists.

This module describes IP named access lists and how to configure them.

- [Finding Feature Information, page 9](#)
- [Information About IP Named Access Control Lists, page 10](#)
- [How to Configure IP Named Access Control Lists, page 14](#)
- [Configuration Examples for IP Named Access Control Lists, page 17](#)
- [Additional References for IP Named Access Control Lists, page 17](#)
- [Feature Information for IP Named Access Control Lists, page 18](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About IP Named Access Control Lists

Definition of an Access List

Access control lists (ACLs) perform packet filtering to control the movement of packets through a network. Packet filtering provides security by limiting the access of traffic into a network, restricting user and device access to a network, and preventing traffic from leaving a network. IP access lists reduce the chance of spoofing and denial-of-service attacks, and allow dynamic, temporary user-access through a firewall.

IP access lists can also be used for purposes other than security, such as to control bandwidth, restrict the content of routing updates, redistribute routes, trigger dial-on-demand (DDR) calls, limit debug output, and identify or classify traffic for quality of service (QoS) features.

An access list is a sequential list that consists of at least one **permit** statement and possibly one or more **deny** statements. In the case of IP access lists, these statements can apply to IP addresses, upper-layer IP protocols, or other fields in IP packets.

Access lists are identified and referenced by a name or a number. Access lists act as packet filters, filtering packets based on the criteria defined in each access list.

After you configure an access list, for the access list to take effect, you must either apply the access list to an interface (by using the **ip access-group** command), a vty (by using the **access-class** command), or reference the access list by any command that accepts an access list. Multiple commands can reference the same access list.

In the following configuration, an IP access list named `branchoffices` is configured on Fast Ethernet interface 0/1/0 and applied to incoming packets. Networks other than the ones specified by the source address and mask pair cannot access Fast Ethernet interface 0/1/0. The destinations for packets coming from sources on network 172.16.7.0 are unrestricted. The destination for packets coming from sources on network 172.16.2.0 must be 172.31.5.4.

```
ip access-list extended branchoffices
 10 permit 172.16.7.0 0.0.0.3 any
 20 permit 172.16.2.0 0.0.0.255 host 172.31.5.4
!
interface fastethernet 0/1/0
 ip access-group branchoffices in
```

Named or Numbered Access Lists

All access lists must be identified by a name or a number. Named access lists are more convenient than numbered access lists because you can specify a meaningful name that is easier to remember and associate with a task. You can reorder statements in or add statements to a named access list.

Named access lists support the following features that are not supported by numbered access lists:

- IP options filtering
- Noncontiguous ports
- TCP flag filtering
- Deleting of entries with the **no permit** or **no deny** command

**Note**

Not all commands that accept a numbered access list will accept a named access list. For example, vty uses only numbered access lists.

Benefits of IP Access Lists

Access control lists (ACLs) perform packet filtering to control the flow of packets through a network. Packet filtering can restrict the access of users and devices to a network, providing a measure of security. Access lists can save network resources by reducing traffic. The benefits of using access lists are as follows:

- Authenticate incoming rsh and rcp requests—Access lists can simplify the identification of local users, remote hosts, and remote users in an authentication database that is configured to control access to a device. The authentication database enables Cisco software to receive incoming remote shell (rsh) and remote copy (rcp) protocol requests.
- Block unwanted traffic or users—Access lists can filter incoming or outgoing packets on an interface, thereby controlling access to a network based on source addresses, destination addresses, or user authentication. You can also use access lists to determine the types of traffic that are forwarded or blocked at device interfaces. For example, you can use access lists to permit e-mail traffic to be routed through a network and to block all Telnet traffic from entering the network.
- Control access to vty—Access lists on an inbound vty (Telnet) can control who can access the lines to a device. Access lists on an outbound vty can control the destinations that the lines from a device can reach.
- Identify or classify traffic for QoS features—Access lists provide congestion avoidance by setting the IP precedence for Weighted Random Early Detection (WRED) and committed access rate (CAR). Access lists also provide congestion management for class-based weighted fair queueing (CBWFQ), priority queueing, and custom queueing.
- Limit debug command output—Access lists can limit debug output based on an IP address or a protocol.
- Provide bandwidth control—Access lists on a slow link can prevent excess traffic on a network.
- Provide NAT control—Access lists can control which addresses are translated by Network Address Translation (NAT).
- Reduce the chance of DoS attacks—Access lists reduce the chance of denial-of-service (DoS) attacks. Specify IP source addresses to control traffic from hosts, networks, or users from accessing your network. Configure the TCP Intercept feature to can prevent servers from being flooded with requests for connection.
- Restrict the content of routing updates—Access lists can control routing updates that are sent, received, or redistributed in networks.
- Trigger dial-on-demand calls—Access lists can enforce dial and disconnect criteria.

Access List Rules

The following rules apply to access lists:

- Only one access list per interface, per protocol, and per direction is allowed.
- An access list must contain at least one **permit** statement or all packets are denied entry into the network.
- The order in which access list conditions or match criteria are configured is important. While deciding whether to forward or block a packet, Cisco software tests the packet against each criteria statement in the order in which these statements are created. After a match is found, no more criteria statements are checked. The same **permit** or **deny** statements specified in a different order can result in a packet being passed under one circumstance and denied in another circumstance.
- If an access list is referenced by a name, but the access list does not exist, all packets pass. An interface or command with an empty access list applied to it permits all traffic into the network.
- Standard access lists and extended access lists cannot have the same name.
- Inbound access lists process packets before the packets are routed to an outbound interface. Inbound access lists that have filtering criteria that deny packet access to a network saves the overhead of routing lookup. Packets that are permitted access to a network based on the configured filtering criteria are processed for routing. For inbound access lists, when you configure a **permit** statement, packets are processed after they are received, and when you configure a **deny** statement, packets are discarded.
- Outbound access lists process packets before they leave the device. Incoming packets are routed to the outbound interface and then processed by the outbound access list. For outbound access lists, when you configure a **permit** statement, packets are sent to the output buffer, and when you configure a **deny** statement, packets are discarded.



Note Outbound access list is not supported in Cisco ASR 900 RSP3 Module.

- An access list can control traffic arriving at a device or leaving a device, but not traffic originating at a device.

Helpful Hints for Creating IP Access Lists

The following tips will help you avoid unintended consequences and help you create more efficient, useful access lists.

- Create the access list before applying it to an interface (or elsewhere), because if you apply a nonexistent access list to an interface and then proceed to configure the access list, the first statement is put into effect, and the implicit **deny** statement that follows could cause you immediate access problems.
- Another reason to configure an access list before applying it is because an interface with an empty access list applied to it permits all traffic.
- All access lists need at least one **permit** statement; otherwise, all packets are denied and no traffic passes.
- Because the software stops testing conditions after it encounters the first match (to either a **permit** or **deny** statement), you will reduce processing time and resources if you put the statements that packets are most likely to match at the beginning of the access list. Place more frequently occurring conditions before less frequent conditions.
- Organize your access list so that more specific references in a network or subnet appear before more general ones.

- Use the statement **permit any any** if you want to allow all other packets not already denied. Using the statement **permit any any** in effect avoids denying all other packets with the implicit deny statement at the end of an access list. Do not make your first access list entry **permit any any** because all traffic will get through; no packets will reach the subsequent testing. In fact, once you specify **permit any any**, all traffic not already denied will get through.
- Although all access lists end with an implicit **deny** statement, we recommend use of an explicit **deny** statement (for example, **deny ip any any**). On most platforms, you can display the count of packets denied by issuing the **show access-list** command, thus finding out more information about who your access list is disallowing. Only packets denied by explicit **deny** statements are counted, which is why the explicit **deny** statement will yield more complete data for you.
- While you are creating an access list or after it is created, you might want to delete an entry.
 - You cannot delete an entry from a numbered access list; trying to do so will delete the entire access list. If you need to delete an entry, you need to delete the entire access list and start over.
 - You can delete an entry from a named access list. Use the **no permit** or **no deny** command to delete the appropriate entry.
- In order to make the purpose of individual statements more scannable and easily understood at a glance, you can write a helpful remark before or after any statement by using the **remark** command.
- If you want to deny access to a particular host or network and find out if someone from that network or host is attempting to gain access, include the **log** keyword with the corresponding **deny** statement so that the packets denied from that source are logged for you.
- This hint applies to the placement of your access list. When trying to save resources, remember that an inbound access list applies the filter conditions before the routing table lookup. An outbound access list applies the filter conditions after the routing table lookup.

Where to Apply an Access List

You can apply access lists to the inbound or outbound interfaces of a device. Applying an access list to an inbound interface controls the traffic that enters the interface and applying an access list to an outbound interface controls the traffic that exits the interface.



Note

Outbound access list is not supported in Cisco ASR 900 RSP3 Module.

When software receives a packet at the inbound interface, the software checks the packet against the statements that are configured for the access list. If the access list permits packets, the software processes the packet. Applying access lists to filter incoming packets can save device resources because filtered packets are discarded before entering the device.

Access lists on outbound interfaces filter packets that are transmitted (sent) out of the interface. You can use the TCP Access Control List (ACL) Splitting feature of the Rate-Based Satellite Control Protocol (RBSCP) on the outbound interface to control the type of packets that are subject to TCP acknowledgment (ACK) splitting on an outbound interface.

You can reference an access list by using a **debug** command to limit the amount of debug logs. For example, based on the filtering or matching criteria of the access list, debug logs can be limited to source or destination addresses or protocols.

You can use access lists to control routing updates, dial-on-demand (DDR), and quality of service (QoS) features.

How to Configure IP Named Access Control Lists

Creating an IP Named Access List

You can create an IP named access list to filter source addresses and destination addresses or a combination of addresses and other IP fields. Named access lists allow you to identify your access lists with an intuitive name.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended *name***
4. **remark *remark***
5. **deny *protocol* [*source source-wildcard*] {**any** | **host** {*address* | *name*} {*destination* [*destination-wildcard*] {**any** | **host** {*address* | *name*} [**log**]**
6. **remark *remark***
7. **permit *protocol* [*source source-wildcard*] {**any** | **host** {*address* | *name*} {*destination* [*destination-wildcard*] {**any** | **host** {*address* | *name*} [**log**]**
8. Repeat Steps 4 through 7 to specify more statements for your access list.
9. **end**
10. **show ip access-lists**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list extended <i>name</i> Example: Device(config)# ip access-list extended acl1	Defines an extended IP access list using a name and enters extended named access list configuration mode.

	Command or Action	Purpose
Step 4	<p>remark <i>remark</i></p> <p>Example: Device(config-ext-nacl)# remark protect server by denying sales access to the acl1 network</p>	<p>(Optional) Adds a description for an access list statement.</p> <ul style="list-style-type: none"> • A remark can precede or follow an IP access list entry. • In this example, the remark command reminds the network administrator that the deny command configured in Step 5 denies the Sales network access to the interface.
Step 5	<p>deny <i>protocol</i> [<i>source source-wildcard</i>] {any host {<i>address</i> <i>name</i>} {<i>destination</i> [<i>destination-wildcard</i>] {any host {<i>address</i> <i>name</i>} [log]</p> <p>Example: Device(config-ext-nacl)# deny ip 192.0.2.0 0.0.255.255 host 192.0.2.10 log</p>	<p>(Optional) Denies all packets that match all conditions specified by the remark.</p>
Step 6	<p>remark <i>remark</i></p> <p>Example: Device(config-ext-nacl)# remark allow TCP from any source to any destination</p>	<p>(Optional) Adds a description for an access list statement.</p> <ul style="list-style-type: none"> • A remark can precede or follow an IP access list entry.
Step 7	<p>permit <i>protocol</i> [<i>source source-wildcard</i>] {any host {<i>address</i> <i>name</i>} {<i>destination</i> [<i>destination-wildcard</i>] {any host {<i>address</i> <i>name</i>} [log]</p> <p>Example: Device(config-ext-nacl)# permit tcp any any</p>	<p>Permits all packets that match all conditions specified by the statement.</p>
Step 8	<p>Repeat Steps 4 through 7 to specify more statements for your access list.</p>	<p>Note All source addresses that are not specifically permitted by a statement are denied by an implicit deny statement at the end of the access list.</p>
Step 9	<p>end</p> <p>Example: Device(config-ext-nacl)# end</p>	<p>Exits extended named access list configuration mode and returns to privileged EXEC mode.</p>
Step 10	<p>show ip access-lists</p> <p>Example: Device# show ip access-lists</p>	<p>Displays the contents of all current IP access lists.</p>

Example:

The following is sample output from the **show ip access-lists** command:

```
Device# show ip access-lists acl1
```

```

Extended IP access list acl1
 permit tcp any 192.0.2.0 255.255.255.255 eq telnet
 deny tcp any any
 deny udp any 192.0.2.0 255.255.255.255 lt 1024
 deny ip any any log

```

Applying an Access List to an Interface

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *type number*
4. **ip access-group** {*access-list-number* | *access-list-name*} {**in** | **out**}
5. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 1/0/2	Specifies an interface and enters interface configuration mode.
Step 4	ip access-group { <i>access-list-number</i> <i>access-list-name</i> } { in out } Example: Device(config-if)# ip access-group acl1 in	Applies the specified access list to the inbound interface. • To filter source addresses, apply the access list to the inbound interface.
Step 5	end Example: Device(config-if)# end	Exits interface configuration mode and returns to privileged EXEC mode.

Configuration Examples for IP Named Access Control Lists

Example: Creating an IP Named Access Control List

```
Device# configure terminal
Device(config)# ip access-list extended acl1
Device(config-ext-nacl)# remark protect server by denying sales access to the acl1 network
Device(config-ext-nacl)# deny ip 192.0.2.0 0.0.255.255 host 192.0.2.10 log
Device(config-ext-nacl)# remark allow TCP from any source to any destination
Device(config-ext-nacl)# permit tcp any any
```

Example: Applying the Access List to an Interface

```
Device# configure terminal
Device(config)# interface GigabitEthernet 1/0/2
Device(config-if)# ip access-group acl1 in
```

Additional References for IP Named Access Control Lists

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
Security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IP Named Access Control Lists

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [http://www.cisco.com/go/featurenavigator](#). An account on Cisco.com is not required.

Table 3: Feature Information for IP Named Access Control Lists

Feature Name	Releases	Feature Information
IP Named Access Control Lists	Cisco IOS XE Release 3.3SE	<p>Access control lists (ACLs) perform packet filtering to control the movement of packets through a network. Packet filtering provides security by limiting traffic into a network, restricting user and device access to a network, and preventing traffic from leaving a network. IP access lists reduce the chance of spoofing and denial-of-service attacks, and allow dynamic, temporary user-access through a firewall.</p> <p>In Cisco IOS XE Release 3.3SE, this feature is supported on Cisco Catalyst 3850 Series Switches and Cisco 5700 Wireless LAN Controllers.</p>



IP Access List Entry Sequence Numbering

The IP Access List Entry Sequence Numbering feature allows you to apply sequence numbers to **permit** or **deny** statements as well as reorder, add, or remove such statements from a named IP access list. The IP Access List Entry Sequence Numbering feature makes revising IP access lists much easier. Prior to this feature, you could add access list entries to the end of an access list only; therefore, needing to add statements anywhere except at the end of a named IP access list required reconfiguring the entire access list.

- [Finding Feature Information, page 19](#)
- [Restrictions for IP Access List Entry Sequence Numbering, page 19](#)
- [Information About IP Access List Entry Sequence Numbering, page 20](#)
- [How to Use Sequence Numbers in an IP Access List, page 24](#)
- [Configuration Examples for IP Access List Entry Sequence Numbering, page 28](#)
- [Additional References, page 30](#)
- [Feature Information for IP Access List Entry Sequence Numbering, page 30](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for IP Access List Entry Sequence Numbering

- This feature does not support dynamic, reflexive, or firewall access lists.
- This feature does not support old-style numbered access lists, which existed before named access lists. Keep in mind that you can name an access list with a number, so numbers are allowed when they are entered in the standard or extended named access list (NACL) configuration mode.

Information About IP Access List Entry Sequence Numbering

Purpose of IP Access Lists

Access lists perform packet filtering to control which packets move through the network and where. Such control can help limit network traffic and restrict the access of users and devices to the network. Access lists have many uses, and therefore many commands accept a reference to an access list in their command syntax. Access lists can be used to do the following:

- Filter incoming packets on an interface.
- Filter outgoing packets on an interface.
- Restrict the contents of routing updates.
- Limit debug output based on an address or protocol.
- Control virtual terminal line access.
- Identify or classify traffic for advanced features, such as congestion avoidance, congestion management, and priority and custom queuing.
- Trigger dial-on-demand routing (DDR) calls.

How an IP Access List Works

An access list is a sequential list consisting of a permit statement and a deny statement that apply to IP addresses and possibly upper-layer IP protocols. The access list has a name by which it is referenced. Many software commands accept an access list as part of their syntax.

An access list can be configured and named, but it is not in effect until the access list is referenced by a command that accepts an access list. Multiple commands can reference the same access list. An access list can control traffic arriving at the device or leaving the device, but not traffic originating at the device.

IP Access List Process and Rules

- The software tests the source or destination address or the protocol of each packet being filtered against the conditions in the access list, one condition (**permit** or **deny** statement) at a time.
- If a packet does not match an access list statement, the packet is then tested against the next statement in the list.
- If a packet and an access list statement match, the rest of the statements in the list are skipped and the packet is permitted or denied as specified in the matched statement. The first entry that the packet matches determines whether the software permits or denies the packet. That is, after the first match, no subsequent entries are considered.
- If the access list denies the address or protocol, the software discards the packet and returns an Internet Control Message Protocol (ICMP) Host Unreachable message.

- If no conditions match, the packet is dropped. This is because each access list ends with an unwritten or implicit **deny** statement. That is, if the packet has not been permitted by the time it was tested against each statement, it is denied.
- Because the software stops testing conditions after the first match, the order of the conditions is critical. The same **permit** or **deny** statements specified in a different order could result in a packet being passed under one circumstance and denied in another circumstance.
- If an access list is referenced by name in a command, but the access list does not exist, all packets pass.
- Only one access list per interface, per protocol, per direction is allowed.
- Inbound access lists process packets arriving at the device. Incoming packets are processed before being routed to an outbound interface. An inbound access list is efficient because it saves the overhead of routing lookups if the packet is to be discarded because it is denied by the filtering tests. If the packet is permitted by the tests, it is then processed for routing. For inbound lists, **permit** means continue to process the packet after receiving it on an inbound interface; **deny** means discard the packet.
- Outbound access lists process packets before they leave the device. Incoming packets are routed to the outbound interface and then processed through the outbound access list. For outbound lists, **permit** means send it to the output buffer; **deny** means discard the packet.

Helpful Hints for Creating IP Access Lists

The following tips will help you avoid unintended consequences and help you create more efficient, useful access lists.

- Create the access list before applying it to an interface (or elsewhere), because if you apply a nonexistent access list to an interface and then proceed to configure the access list, the first statement is put into effect, and the implicit **deny** statement that follows could cause you immediate access problems.
- Another reason to configure an access list before applying it is because an interface with an empty access list applied to it permits all traffic.
- All access lists need at least one **permit** statement; otherwise, all packets are denied and no traffic passes.
- Because the software stops testing conditions after it encounters the first match (to either a **permit** or **deny** statement), you will reduce processing time and resources if you put the statements that packets are most likely to match at the beginning of the access list. Place more frequently occurring conditions before less frequent conditions.
- Organize your access list so that more specific references in a network or subnet appear before more general ones.
- Use the statement **permit any any** if you want to allow all other packets not already denied. Using the statement **permit any any** in effect avoids denying all other packets with the implicit deny statement at the end of an access list. Do not make your first access list entry **permit any any** because all traffic will get through; no packets will reach the subsequent testing. In fact, once you specify **permit any any**, all traffic not already denied will get through.
- Although all access lists end with an implicit **deny** statement, we recommend use of an explicit **deny** statement (for example, **deny ip any any**). On most platforms, you can display the count of packets denied by issuing the **show access-list** command, thus finding out more information about who your access list is disallowing. Only packets denied by explicit **deny** statements are counted, which is why the explicit **deny** statement will yield more complete data for you.

- While you are creating an access list or after it is created, you might want to delete an entry.
 - You cannot delete an entry from a numbered access list; trying to do so will delete the entire access list. If you need to delete an entry, you need to delete the entire access list and start over.
 - You can delete an entry from a named access list. Use the **no permit** or **no deny** command to delete the appropriate entry.
- In order to make the purpose of individual statements more scannable and easily understood at a glance, you can write a helpful remark before or after any statement by using the **remark** command.
- If you want to deny access to a particular host or network and find out if someone from that network or host is attempting to gain access, include the **log** keyword with the corresponding **deny** statement so that the packets denied from that source are logged for you.
- This hint applies to the placement of your access list. When trying to save resources, remember that an inbound access list applies the filter conditions before the routing table lookup. An outbound access list applies the filter conditions after the routing table lookup.

Source and Destination Addresses

Source and destination address fields in an IP packet are two typical fields on which to base an access list. Specify source addresses to control the packets being sent from certain networking devices or hosts. Specify destination addresses to control the packets being sent to certain networking devices or hosts.

Wildcard Mask and Implicit Wildcard Mask

When comparing the address bits in an access list entry to a packet being submitted to the access list, address filtering uses wildcard masking to determine whether to check or ignore the corresponding IP address bits. By carefully setting wildcard masks, an administrator can select one or more IP addresses for permit or deny tests.

Wildcard masking for IP address bits uses the number 1 and the number 0 to specify how the software treats the corresponding IP address bits. A wildcard mask is sometimes referred to as an inverted mask because a 1 and 0 mean the opposite of what they mean in a subnet (network) mask.

- A wildcard mask bit 0 means check the corresponding bit value.
- A wildcard mask bit 1 means ignore that corresponding bit value.

If you do not supply a wildcard mask with a source or destination address in an access list statement, the software assumes a default wildcard mask of 0.0.0.0.

Unlike subnet masks, which require contiguous bits indicating network and subnet to be ones, wildcard masks allow noncontiguous bits in the mask.

Transport Layer Information

You can filter packets based on transport layer information, such as whether the packet is a TCP, UDP, Internet Control Message Protocol (ICMP) or Internet Group Management Protocol (IGMP) packet.

Benefits IP Access List Entry Sequence Numbering

The ability to apply sequence numbers to IP access list entries simplifies access list changes. Prior to the IP Access List Entry Sequence Numbering feature, there was no way to specify the position of an entry within an access list. If you wanted to insert an entry (statement) in the middle of an existing list, all of the entries *after* the desired position had to be removed. Then, once you added the new entry, you needed to reenter all of the entries you removed earlier. This method was cumbersome and error prone.

The IP Access List Entry Sequence Numbering feature allows you to add sequence numbers to access list entries and resequence them. When you add a new entry, you can choose the sequence number so that the entry is in a desired position in the access list. If necessary, entries currently in the access list can be resequenced (reordered) to create room to insert the new entry.

Sequence Numbering Behavior

- For backward compatibility with previous releases, if entries with no sequence numbers are applied, the first entry is assigned a sequence number of 10, and successive entries are incremented by 10. The maximum sequence number is 2147483647. If the generated sequence number exceeds this maximum number, the following message is displayed:

```
Exceeded maximum sequence number.
```

- If you enter an entry without a sequence number, it is assigned a sequence number that is 10 greater than the last sequence number in that access list and is placed at the end of the list.
- If you enter an entry that matches an already existing entry (except for the sequence number), then no changes are made.
- If you enter a sequence number that is already present, the following error message is generated:

```
Duplicate sequence number.
```

- If a new access list is entered from global configuration mode, then sequence numbers for that access list are generated automatically.
- Distributed support is provided so that the sequence numbers of entries in the Route Processor (RP) and line card (LC) are always synchronized.
- Sequence numbers are not nvgened. That is, the sequence numbers themselves are not saved. In the event that the system is reloaded, the configured sequence numbers revert to the default sequence starting number and increment from that number. The function is provided for backward compatibility with software releases that do not support sequence numbering.
- The IP Access List Entry Sequence Numbering feature works with named standard and extended IP access lists. Because the name of an access list can be designated as a number, numbers are acceptable.

How to Use Sequence Numbers in an IP Access List

Sequencing Access-List Entries and Revising the Access List

This task shows how to assign sequence numbers to entries in a named IP access list and how to add or delete an entry to or from an access list. When completing this task, keep the following points in mind:

- Resequencing the access list entries is optional. The resequencing step in this task is shown as required because that is one purpose of this feature and this task demonstrates that functionality.
- In the following procedure, the **permit** command is shown in Step 5 and the **deny** command is shown in Step 6. However, that order can be reversed. Use the order that suits the need of your configuration.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list resequence** *access-list-name starting-sequence-number increment*
4. **ip access-list {standard| extended}** *access-list-name*
5. Do one of the following:
 - *sequence-number* **permit** *source source-wildcard*
 - *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
6. Do one of the following:
 - *sequence-number* **deny** *source source-wildcard*
 - *sequence-number* **deny** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
7. Do one of the following:
 - *sequence-number* **permit** *source source-wildcard*
 - *sequence-number* **permit** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
8. Do one of the following:
 - *sequence-number* **deny** *source source-wildcard*
 - *sequence-number* **deny** *protocol source source-wildcard destination destination-wildcard* [**precedence** *precedence*][**tos** *tos*] [**log**] [**time-range** *time-range-name*] [**fragments**]
9. Repeat Step 5 and/or Step 6 to add sequence number statements, as applicable.
10. **end**
11. **show ip access-lists** *access-list-name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. Enter your password if prompted.

	Command or Action	Purpose
Step 2	<p>configure terminal</p> <p>Example:</p> <pre>Device# configure terminal</pre>	Enters global configuration mode.
Step 3	<p>ip access-list resequence <i>access-list-name</i> <i>starting-sequence-number</i> <i>increment</i></p> <p>Example:</p> <pre>Device(config)# ip access-list resequence kmdl 100 15</pre>	Resequences the specified IP access list using the starting sequence number and the increment of sequence numbers.
Step 4	<p>ip access-list {standard extended} <i>access-list-name</i></p> <p>Example:</p> <pre>Device(config)# ip access-list standard kmdl</pre>	<p>Specifies the IP access list by name and enters named access list configuration mode.</p> <ul style="list-style-type: none"> • If you specify standard, make sure you subsequently specify permit and/or deny statements using the standard access list syntax. • If you specify extended, make sure you subsequently specify permit and/or deny statements using the extended access list syntax.
Step 5	<p>Do one of the following:</p> <ul style="list-style-type: none"> • <i>sequence-number</i> permit <i>source</i> <i>source-wildcard</i> • <i>sequence-number</i> permit <i>protocol</i> <i>source</i> <i>source-wildcard</i> <i>destination</i> <i>destination-wildcard</i> [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] <p>Example:</p> <pre>Device(config-std-nacl)# 105 permit 10.5.5.5 0.0.0 255</pre>	<p>Specifies a permit statement in named IP access list mode.</p> <ul style="list-style-type: none"> • This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • As the prompt indicates, this access list was a standard access list. If you had specified extended in Step 4, the prompt for this step would be <code>Device(config-ext-nacl)</code> and you would use the extended permit command syntax.
Step 6	<p>Do one of the following:</p> <ul style="list-style-type: none"> • <i>sequence-number</i> deny <i>source</i> <i>source-wildcard</i> • <i>sequence-number</i> deny <i>protocol</i> <i>source</i> <i>source-wildcard</i> <i>destination</i> <i>destination-wildcard</i> [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] 	<p>(Optional) Specifies a deny statement in named IP access list mode.</p> <ul style="list-style-type: none"> • This access list uses a permit statement first, but a deny statement could appear first, depending on the order of statements you need. • As the prompt indicates, this access list was a standard access list. If you had specified extended in Step 4, the prompt for this step would be <code>Device(config-ext-nacl)</code> and you would use the extended deny command syntax.

	Command or Action	Purpose
	<p>Example:</p> <pre>Device(config-std-nacl)# 105 deny 10.6.6.7 0.0.0 255</pre>	
Step 7	<p>Do one of the following:</p> <ul style="list-style-type: none"> <i>sequence-number</i> permit <i>source source-wildcard</i> <i>sequence-number</i> permit <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] <p>Example:</p> <pre>Device(config-ext-nacl)# 150 permit tcp any any log</pre>	<p>Specifies a permit statement in named IP access list mode.</p> <ul style="list-style-type: none"> This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. See the permit (IP) command for additional command syntax to permit upper layer protocols (ICMP, IGMP, TCP, and UDP). Use the no <i>sequence-number</i> command to delete an entry.
Step 8	<p>Do one of the following:</p> <ul style="list-style-type: none"> <i>sequence-number</i> deny <i>source source-wildcard</i> <i>sequence-number</i> deny <i>protocol source source-wildcard destination destination-wildcard</i> [precedence <i>precedence</i>][tos <i>tos</i>] [log] [time-range <i>time-range-name</i>] [fragments] <p>Example:</p> <pre>Device(config-ext-nacl)# 150 deny tcp any any log</pre>	<p>(Optional) Specifies a deny statement in named IP access list mode.</p> <ul style="list-style-type: none"> This access list happens to use a permit statement first, but a deny statement could appear first, depending on the order of statements you need. See the deny (IP) command for additional command syntax to permit upper layer protocols (ICMP, IGMP, TCP, and UDP). Use the no <i>sequence-number</i> command to delete an entry.
Step 9	Repeat Step 5 and/or Step 6 to add sequence number statements, as applicable.	Allows you to revise the access list.
Step 10	<p>end</p> <p>Example:</p> <pre>Device(config-std-nacl)# end</pre>	(Optional) Exits the configuration mode and returns to privileged EXEC mode.
Step 11	<p>show ip access-lists <i>access-list-name</i></p> <p>Example:</p> <pre>Device# show ip access-lists kmdl</pre>	(Optional) Displays the contents of the IP access list.

Examples

Review the output of the **show ip access-lists** command to see that the access list includes the new entries:

```
Device# show ip access-lists kmd1

Standard IP access list kmd1
100 permit 10.4.4.0, wildcard bits 0.0.0.255
105 permit 10.5.5.0, wildcard bits 0.0.0.255
115 permit 10.0.0.0, wildcard bits 0.0.0.255
130 permit 10.5.5.0, wildcard bits 0.0.0.255
145 permit 10.0.0.0, wildcard bits 0.0.0.255
```

Configuration Examples for IP Access List Entry Sequence Numbering

Example: Resequencing Entries in an Access List

The following example shows access list resequencing. The starting value is 1, and increment value is 2. The subsequent entries are ordered based on the increment values specified, and the range is from 1 to 2147483647.

When an entry with no sequence number is entered, by default the entry has a sequence number of 10 more than the last entry in the access list.

```
Device# show access-list 150

Extended IP access list 150
 10 permit ip host 10.3.3.3 host 172.16.5.34
 20 permit icmp any any
 30 permit tcp any host 10.3.3.3
 40 permit ip host 10.4.4.4 any
 50 Dynamic test permit ip any any
 60 permit ip host 172.16.2.2 host 10.3.3.12
 70 permit ip host 10.3.3.3 any log
 80 permit tcp host 10.3.3.3 host 10.1.2.2
 90 permit ip host 10.3.3.3 any
100 permit ip any any

Device(config)# ip access-list extended 150
Device(config)# ip access-list resequence 150 1 2
Device(config)# exit

Device# show access-list 150

Extended IP access list 150
 1 permit ip host 10.3.3.3 host 172.16.5.34
 3 permit icmp any any
10 permit tcp any any eq 22 log
 5 permit tcp any host 10.3.3.3
 7 permit ip host 10.4.4.4 any
 9 Dynamic test permit ip any any
11 permit ip host 172.16.2.2 host 10.3.3.12
13 permit ip host 10.3.3.3 any log
15 permit tcp host 10.3.3.3 host 10.1.2.2
17 permit ip host 10.3.3.3 any
19 permit ip any any
```

Example: Adding Entries with Sequence Numbers

In the following example, a new entry is added to a specified access list:

```
Device# show ip access-list

Standard IP access list tryon
 2 permit 10.4.4.2, wildcard bits 0.0.255.255
 5 permit 10.0.0.44, wildcard bits 0.0.0.255
10 permit 10.0.0.1, wildcard bits 0.0.0.255
20 permit 10.0.0.2, wildcard bits 0.0.0.255

Device(config)# ip access-list standard tryon
Device(config-std-nacl)# 15 permit 10.5.5.5 0.0.0.255
Device(config-std-nacl)# exit
Device(config)# exit
Device# show ip access-list

Standard IP access list tryon
 2 permit 10.4.0.0, wildcard bits 0.0.255.255
 5 permit 10.0.0.0, wildcard bits 0.0.0.255
10 permit 10.0.0.0, wildcard bits 0.0.0.255
15 permit 10.5.5.0, wildcard bits 0.0.0.255
20 permit 10.0.0.0, wildcard bits 0.0.0.255
```

Example: Entry Without Sequence Number

The following example shows how an entry with no specified sequence number is added to the end of an access list. When an entry is added without a sequence number, it is automatically given a sequence number that puts it at the end of the access list. Because the default increment is 10, the entry will have a sequence number 10 higher than the last entry in the existing access list.

```
Device(config)# ip access-list standard 1
Device(config-std-nacl)# permit 10.1.1.1 0.0.0.255
Device(config-std-nacl)# permit 10.2.2.2 0.0.0.255
Device(config-std-nacl)# permit 10.3.3.3 0.0.0.255
Device(config-std-nacl)## exit
Device# show access-list

Standard IP access list 1
10 permit 0.0.0.0, wildcard bits 0.0.0.255
20 permit 0.0.0.0, wildcard bits 0.0.0.255
30 permit 0.0.0.0, wildcard bits 0.0.0.255

Device(config)# ip access-list standard 1
Device(config-std-nacl)# permit 10.4.4.4 0.0.0.255
Device(config-std-nacl)# end
Device(config-std-nacl)## exit
Device# show access-list

Standard IP access list 1
10 permit 0.0.0.0, wildcard bits 0.0.0.255
20 permit 0.0.0.0, wildcard bits 0.0.0.255
30 permit 0.0.0.0, wildcard bits 0.0.0.255
40 permit 0.0.0.0, wildcard bits 0.0.0.255
```

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases
IP access list commands	<i>Cisco IOS Security Command Reference</i>
Configuring IP access lists	“Creating an IP Access List and Applying It to an Interface”

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IP Access List Entry Sequence Numbering

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [http://www.cisco.com/go/featurenavigator](#). An account on Cisco.com is not required.

Table 4: Feature Information for IP Access List Entry Sequence Numbering

Feature Name	Releases	Feature Information
IP Access List Entry Sequence Numbering	Cisco IOS XE Release 3.3SE	<p>Users can apply sequence numbers to permit or deny statements and also reorder, add, or remove such statements from a named IP access list. This feature makes revising IP access lists much easier. Prior to this feature, users could add access list entries to the end of an access list only; therefore needing to add statements anywhere except the end required reconfiguring the access list entirely.</p> <p>In , , support was added for the Cisco Catalyst 3850 Series Switches.</p> <p>In Cisco IOS XE Release 3.3SE, this feature is supported on Cisco Catalyst 3850 Series Switches and Cisco 5700 Wireless LAN Controllers.</p> <p>The following commands were introduced or modified: deny (IP), ip access-list resequence deny (IP), permit (IP).</p>



Standard IP Access List Logging

The Standard IP Access List Logging feature provides the ability to log messages about packets that are permitted or denied by a standard IP access list. Any packet that matches the access list logs an information message about the packet at the device console.

This module provides information about standard IP access list logging.

- [Finding Feature Information, page 33](#)
- [Restrictions for Standard IP Access List Logging, page 33](#)
- [Information About Standard IP Access List Logging, page 34](#)
- [How to Configure Standard IP Access List Logging, page 34](#)
- [Configuration Examples for Standard IP Access List Logging, page 37](#)
- [Additional References for Standard IP Access List Logging, page 37](#)
- [Feature Information for Standard IP Access List Logging, page 38](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Restrictions for Standard IP Access List Logging

IP access list logging is supported only for routed interfaces or router access control lists (ACLs).

Information About Standard IP Access List Logging

Standard IP Access List Logging

The Standard IP Access List Logging feature provides the ability to log messages about packets that are permitted or denied by a standard IP access list. Any packet that matches the access list causes an information log message about the packet to be sent to the device console. The log level of messages that are printed to the device console is controlled by the **logging console** command.

The first packet that the access list inspects triggers the access list to log a message at the device console. Subsequent packets are collected over 5-minute intervals before they are displayed or logged. Log messages include information about the access list number, the source IP address of packets, the number of packets from the same source that were permitted or denied in the previous 5-minute interval, and whether a packet was permitted or denied. You can also monitor the number of packets that are permitted or denied by a particular access list, including the source address of each packet.

How to Configure Standard IP Access List Logging

Creating a Standard IP Access List Using Numbers

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* {deny | permit} *host address* [log]
4. **access-list** *access-list-number* {deny | permit} **any** [log]
5. **interface** *type number*
6. **ip access-group** *access-list-number* {in | out}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. <ul style="list-style-type: none"> • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	access-list <i>access-list-number</i> {deny permit} host <i>address</i> [log] Example: Device(config)# access-list 1 permit host 10.1.1.1 log	Defines a standard numbered IP access list using a source address and wildcard, and configures the logging of informational messages about packets that match the access list entry at the device console.
Step 4	access-list <i>access-list-number</i> {deny permit} any [log] Example: Device(config)# access-list 1 permit any log	Defines a standard numbered IP access list by using an abbreviation for the source and source mask 0.0.0.0 255.255.255.255.
Step 5	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 0/0	Configures an interface and enters interface configuration mode.
Step 6	ip access-group <i>access-list-number</i> {in out} Example: Device(config-if)# ip access-group 1 in	Applies the specified numbered access list to the incoming or outgoing interface. <ul style="list-style-type: none"> • When you filter based on source addresses, you typically apply the access list to an incoming interface.
Step 7	end Example: Device(config-if)# end	Exits interface configuration mode and enters privileged EXEC mode.

Creating a Standard IP Access List Using Names

SUMMARY STEPS

1. enable
2. configure terminal
3. ip access-list standard *name*
4. {deny | permit} {host *address* | any} log
5. exit
6. interface *type number*
7. ip access-group *access-list-name* {in | out}
8. end

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ip access-list standard <i>name</i> Example: Device(config)# ip access-list standard acl1	Defines a standard IP access list and enters standard named access list configuration mode.
Step 4	{deny permit} {host <i>address</i> any} log Example: Device(config-std-nacl)# permit host 10.1.1.1 log	Sets conditions in a named IP access list that will deny packets from entering a network or permit packets to enter a network, and configures the logging of informational messages about packets that match the access list entry at the device console.
Step 5	exit Example: Device(config-std-nacl)# exit	Exits standard named access list configuration mode and enters global configuration mode.
Step 6	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 0/0	Configures an interface and enters interface configuration mode.
Step 7	ip access-group <i>access-list-name</i> {in out} Example: Device(config-if)# ip access-group acl1 in	Applies the specified access list to the incoming or outgoing interface. • When you filter based on source addresses, you typically apply the access list to an incoming interface.
Step 8	end Example: Device(config-if)# end	Exits interface configuration mode and enters privileged EXEC mode.

Configuration Examples for Standard IP Access List Logging

Example: Creating a Standard IP Access List Using Numbers

```
Device# configure terminal
Device(config)# access-list 1 permit host 10.1.1.1 log
Device(config)# access-list 1 permit any log
Device(config)# interface GigabitEthernet 0/0
Device(config-if)# ip access-group 1 in
```

Example: Creating a Standard IP Access List Using Names

```
Device# configure terminal
Device(config)# ip access-list standard acl1
Device(config-std-nacl)# permit host 10.1.1.1 log
Device(config-std-nacl)# exit
Device(config)# interface GigabitEthernet 0/0
Device(config-if)# ip access-group acl1 in
```

Example: Limiting Debug Output

The following sample configuration uses an access list to limit the **debug** command output. Limiting the **debug** output restricts the volume of data to what you are interested in, saving you time and resources.

```
Device(config)# ip access-list acl1
Device(config-std-nacl)# remark Displays only advertisements for LDP peer in acl1
Device(config-std-nacl)# permit host 10.0.0.44

Device# debug mpls ldp advertisements peer-acl acl1

tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.17.0.33
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.16.0.31
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 172.22.0.33
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.0.1
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.0.3
tagcon: peer 10.0.0.44:0 (pp 0x60E105BC): advertise 192.168.1.33
```

Additional References for Standard IP Access List Logging

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases

Related Topic	Document Title
Security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for Standard IP Access List Logging

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [http://www.cisco.com/go/featurenavigator](#). An account on Cisco.com is not required.

Table 5: Feature Information for Standard IP Access List Logging

Feature Name	Releases	Feature Information
Standard IP Access List Logging	Cisco IOS XE Release 3.3SE	<p>The Standard IP Access List Logging feature provides the ability to log messages about packets that are permitted or denied by a standard IP access list. Any packet that matches the access list logs an information message about the packet at the device console.</p> <p>In Cisco IOS XE Release 3.3SE, this feature is supported on Cisco Catalyst 3850 Series Switches and Cisco 5700 Wireless LAN Controllers.</p>



IPv6 PACL Support

The IPv6 PACL feature permits or denies the movement of traffic between Layer 3 subnets and VLANs, or within a VLAN.

- [Finding Feature Information, page 39](#)
- [Prerequisites for IPv6 PACL Support, page 39](#)
- [Information About IPv6 PACL Support, page 40](#)
- [How to Configure IPv6 PACL Support, page 40](#)
- [Configuration Examples for IPv6 PACL Support, page 41](#)
- [Additional References for IPv6 PACL Support, page 41](#)
- [Feature Information for IPv6 PACL Support, page 42](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for IPv6 PACL Support

In order to use the IPv6 port-based access control list (PACL) feature, you must know how to configure IPv6 access lists.

Information About IPv6 PACL Support

IPv6 Port-Based Access Control List Support

The IPv6 PACL feature provides the ability to provide access control (permit or deny) on Layer 2 switch ports for IPv6 traffic. IPv6 PACLs are similar to IPv4 PACLs, which provide access control on Layer 2 switch ports for IPv4 traffic. They are supported only in the ingress direction and in hardware.

A PACL can filter ingress traffic on Layer 2 interfaces based on Layer 3 and Layer 4 header information or non-IP Layer 2 information.

How to Configure IPv6 PACL Support

Configuring PACL Mode and Applying IPv6 PACL on an Interface

Before You Begin

Before you configure the IPv6 PACL feature, you must configure an IPv6 access list. Once you have configured the IPv6 access list, you must configure the port-based access control list (PACL) mode on the specified IPv6 Layer 2 interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. **exit**
5. **interface** *type number*
6. **ipv6 traffic-filter** *access-list-name* {**in** | **out**}
7. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.

	Command or Action	Purpose
Step 3	ipv6 access-list <i>access-list-name</i> Example: Device(config)# ipv6 access-list list1	Defines an IPv6 ACL and enters IPv6 access list configuration mode.
Step 4	exit Example: Device(config-ipv6-acl)# exit	Exits IPv6 access list configuration mode and enters global configuration mode.
Step 5	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 0/0	Specifies an interface type and number and enters interface configuration mode.
Step 6	ipv6 traffic-filter <i>access-list-name</i> { in out } Example: Device(config-if)# ipv6 traffic-filter list1 in	Filters incoming and outgoing IPv6 traffic on an interface.
Step 7	end Example: Device(config-if)# end	Exits interface configuration mode and enters privileged EXEC mode.

Configuration Examples for IPv6 PACL Support

Example: Configuring PACL Mode and Applying IPv6 PACL on an Interface

```
Device# configure terminal
Device(config)# ipv6 access-list list1
Device(config-ipv6-acl)# exit
Device(config)# interface GigabitEthernet 0/0
Device(config-if)# ipv6 traffic-filter list1 in
```

Additional References for IPv6 PACL Support

Related Documents

Related Topic	Document Title
IPv6 addressing and connectivity	<i>IPv6 Configuration Guide</i>

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
IPv6 commands	<i>Cisco IOS IPv6 Command Reference</i>
Cisco IOS IPv6 features	Cisco IOS IPv6 Feature Mapping

Standards and RFCs

Standard/RFC	Title
RFCs for IPv6	<i>IPv6 RFCs</i>

MIBs

MIB	MIBs Link
No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature.	To locate and download MIBs for selected platforms, Cisco IOS releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 PACL Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [http://www.cisco.com/go/featurenavigator](#). An account on Cisco.com is not required.

Table 6: Feature Information for IPv6 PACL Support

Feature Name	Releases	Feature Information
IPv6 PACL Support	Cisco IOS XE Release 3.3SE	<p>The IPv6 PACL feature permits or denies the movement of traffic between port-based interface, Layer 3 subnets, wireless or wired clients, and VLANs, or within a VLAN.</p> <p>In Cisco IOS XE Release 3.3SE, this feature is supported on Cisco Catalyst 3850 Series Switches and Cisco 5700 Wireless LAN Controllers.</p> <p>The following command was introduced or modified: ipv6 traffic-filter.</p>



CHAPTER 6

IPv6 Services—Standard Access Control Lists

Access lists determine the type of traffic that is blocked or forwarded at device interfaces. Access control lists (ACLs) allow the filtering of inbound and outbound traffic at interfaces based on source and destination addresses.

This module provides information about standard IPv6 ACLs.

- [Finding Feature Information, page 45](#)
- [Information About IPv6 Services--Standard Access Control Lists, page 45](#)
- [How to Configure IPv6 Services--Standard Access Control Lists, page 46](#)
- [Configuration Examples for IPv6 Services--Standard Access Control Lists, page 49](#)
- [Additional References for IPv6 Services—Standard Access Control Lists, page 49](#)
- [Feature Information for IPv6 Services—Standard Access Control Lists, page 50](#)

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the feature information table.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Information About IPv6 Services--Standard Access Control Lists

Access Control Lists for IPv6 Traffic Filtering

The standard ACL functionality in IPv6 is similar to standard ACLs in IPv4. Access lists determine what traffic is blocked and what traffic is forwarded at device interfaces and allow filtering based on source and destination addresses, inbound and outbound to a specific interface. Each access list has an implicit deny

statement at the end. IPv6 ACLs are defined and their deny and permit conditions are set using the **ipv6 access-list** command with the **deny** and **permit** keywords in global configuration mode.

IPv6 extended ACLs augments standard IPv6 ACL functionality to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control (functionality similar to extended ACLs in IPv4).

IPv6 Packet Inspection

The following header fields are used for IPv6 inspection: traffic class, flow label, payload length, next header, hop limit, and source or destination IP address. For further information on and descriptions of the IPv6 header fields, see RFC 2474.

Access Class Filtering in IPv6

Filtering incoming and outgoing connections to and from the device based on an IPv6 ACL is performed using the **ipv6 access-class** command in line configuration mode. The **ipv6 access-class** command is similar to the **access-class** command, except the IPv6 ACLs are defined by a name. If the IPv6 ACL is applied to inbound traffic, the source address in the ACL is matched against the incoming connection source address and the destination address in the ACL is matched against the local device address on the interface. If the IPv6 ACL is applied to outbound traffic, the source address in the ACL is matched against the local device address on the interface and the destination address in the ACL is matched against the outgoing connection source address. We recommend that identical restrictions are set on all the virtual terminal lines because a user can attempt to connect to any of them.

How to Configure IPv6 Services--Standard Access Control Lists

Configuring IPv6 Services—Standard Access Control Lists

IPv6 access control lists (ACLs) do not contain implicit permit rules.

The IPv6 neighbor discovery process uses the IPv6 network-layer service; therefore, you must configure IPv6 ACLs to allow IPv6 neighbor discovery packets to be sent and received on an interface.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. **permit** *source-ipv6-prefix/prefix-length* **host**
5. **deny** *protocol source-ipv6-prefix/prefix-length* **eq telnet any**
6. **exit**
7. **interface** *type number*
8. **ipv6 traffic-filter** *access-list-name* {**in** | **out**}
9. **end**
10. **show ipv6 access-list** [*access-list-name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	enable Example: Device> enable	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	configure terminal Example: Device# configure terminal	Enters global configuration mode.
Step 3	ipv6 access-list <i>access-list-name</i> Example: Device(config)# ipv6 access-list ipv6acl	Defines an IPv6 ACL and enters IPv6 access list configuration mode. • IPv6 ACL names cannot contain a space or quotation mark or begin with a numeral.
Step 4	permit <i>source-ipv6-prefix/prefix-length</i> host Example: Device(config-ipv6-acl)# permit 2001:DB8:1::1/32 any	Specifies permit conditions for the IPv6 access list.
Step 5	deny <i>protocol source-ipv6-prefix/prefix-length</i> eq telnet any Example: Device(config-ipv6-acl)# deny tcp 2001:DB8:0300:0201::/32 eq telnet any	Specifies deny conditions for the IPv6 access list.

	Command or Action	Purpose
Step 6	exit Example: Device(config-ipv6-acl)# exit	Exits IPv6 access list configuration mode and enters global configuration mode.
Step 7	interface <i>type number</i> Example: Device(config)# interface GigabitEthernet 1/0/2	Configures an interface and enters interface configuration mode.
Step 8	ipv6 traffic-filter <i>access-list-name</i> { in out } Example: Device(config-if)# ipv6 traffic-filter ipv6acl out	Applies the specified IPv6 access list to the interface configured in Step 7.
Step 9	end Example: Device(config-if)# end	Exits interface configuration mode and enters privileged EXEC mode.
Step 10	show ipv6 access-list [<i>access-list-name</i>] Example: Device# show ipv6 access-list	Displays the contents of all current IPv6 access lists.

Example:

The following is sample output from the **show ipv6 access-list** command:

```
Device# show ipv6 access-list

IPv6 access list ipv6acl
  permit tcp any any eq bgp reflect tcptraffic (8 matches) sequence 10
  permit tcp any any eq telnet reflect tcptraffic (15 matches) sequence 20
  permit udp any any reflect udptraffic sequence 30

IPv6 access list tcptraffic (reflexive) (per-user)
  permit tcp host 2001:DB8:1::32 eq bgp host 2001:DB8:2::32 eq 11000 timeout 300 (time
left 243) sequence 1
  permit tcp host 2001:DB8:1::32 eq telnet host 2001:DB8:2::32 eq 11001 timeout 300 (time
left 296) sequence 2

IPv6 access list outbound
  evaluate udptraffic
  evaluate tcptraffic
```

Configuration Examples for IPv6 Services--Standard Access Control Lists

Example: Configuring IPv6 Services—Standard Access Control Lists

```
Device# configure terminal
Device(config)# ipv6 access-list ipv6acl
Device(config-ipv6-acl)# permit 2001:DB8:1::1/32 any
Device(config-ipv6-acl)# deny tcp 2001:DB8:0300:0201::/32 eq telnet any
Device(config-ipv6-acl)# exit
Device(config)# interface GigabitEthernet 1/0/2
Device(config-if)# ipv6 traffic-filter ipv6acl out
Device(config-if)# end
```

Example: Creating and Applying an IPv6 ACL

The following example shows how to restrict HTTP access to certain hours during the day and log any activity outside of the permitted hours:

```
Device# configure terminal
Device(config)# time-range lunchtime
Device(config-time-range)# periodic weekdays 12:00 to 13:00
Device(config-time-range)# exit
Device(config)# ipv6 access-list OUTBOUND
Device(config-ipv6-acl)# permit tcp any any eq www time-range lunchtime
Device(config-ipv6-acl)# deny tcp any any eq www log-input
Device(config-ipv6-acl)# permit tcp 2001:DB8::/32 any
Device(config-ipv6-acl)# permit udp 2001:DB8::/32 any
Device(config-ipv6-acl)# end
```

Additional References for IPv6 Services—Standard Access Control Lists

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Command List, All Releases

Related Topic	Document Title
Security commands	<ul style="list-style-type: none"> • Cisco IOS Security Command Reference: Commands A to C • Cisco IOS Security Command Reference: Commands D to L • Cisco IOS Security Command Reference: Commands M to R • Cisco IOS Security Command Reference: Commands S to Z
IPv6 Commands	Cisco IOS IPv6 Command Reference

Standards and RFCs

Standard/RFC	Title
RFC 2474	Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers

Technical Assistance

Description	Link
The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.	http://www.cisco.com/cisco/web/support/index.html

Feature Information for IPv6 Services—Standard Access Control Lists

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [http://www.cisco.com/cisco/web/featurenavigator/index.html](#). An account on Cisco.com is not required.

Table 7: Feature Information for IPv6 Services—Standard Access Control Lists

Feature Name	Releases	Feature Information
IPv6 Services—Standard Access Control Lists	Cisco IOS XE Release 3.3SE	<p>Access lists determine the type of traffic that is blocked or forwarded at device interfaces. Access control lists (ACLs) allow the filtering of inbound and outbound traffic based on source and destination addresses at interfaces. Standard IPv6 ACLs support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information.</p> <p>In Cisco IOS XE Release 3.3SE, this feature is supported on Cisco Catalyst 3850 Series Switches and Cisco 5700 Wireless LAN Controllers</p> <p>The following commands were introduced or modified: ipv6 access-list, show ipv6 access-list.</p>

