IPv6 Access Control Lists

Access lists determine what traffic is blocked and what traffic is forwarded at device interfaces and allow filtering of traffic based on source and destination addresses, and inbound and outbound traffic to a specific interface. Standard IPv6 ACL functionality was extended to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control. Standard IPv6 ACL functionality was extended to support traffic filtering based on IPv6 option headers and optional, upper-layer protocol type information for finer granularity of control.

This module describes how to configure IPv6 traffic filtering and to control access to virtual terminal lines.

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 https://cfnng.cisco.com/. An account on Cisco.com is not required.

Restrictions for IPv6 Access Control Lists

- IPv4 / IPv6 ACL is supported only at the access Layer 3 or bridge domain interfaces. It is not supported on the MPLS enabled core interfaces.
- IPv6 ACL is *not* supported on port-channel member link.
- IPv4 or IPv6 is *not* supported on label interface.
- MAC ACL supports only non-IP packets.

- MAC ACL counters are *not* supported.
- MAC ACL is *not* supported on trunk EFP or port channel EFP.
- TCAMS are programmed for ACL configured on both physical interface and BDI.
- A total of 512 unique and combination of IPv4 and IPv6 ACLs can be configured.
- Filtering based on extension header types is *not* supported except for filtering based on fragmentation.
- Filtering based on IPv6 Protocols and L4 ports will *not* work for IPv6 traffic carrying extension headers.
- Filtering based on TCP Control Flags is *not* supported.
- Egress IPv6 ACL is *not* supported.
- All NDP packets (ICMPv6 type 133-137) are allowed by default.
- IPv6 ACL is *not* supported on Link Local Addresses (LL).

Note IPv6 packets with LL address as SA is permitted by default.

- Per ACE counters are *not* supported.
- Re-sequencing of ACE's for an IPv6 ACL is *not* supported.
- Starting Cisco IOS XE Release 3.18SP, IPv6 ACLs are supported on the RSP3 module.
- Although the hardware is not programmed for TCAM exhaustion, ACL stats get incremented for control packets on the interface.

Restrictions - Dual Stack (IPv4 + IPv6 ACL on an Interface)

- When a unique combination of IPv4 ACL and IPv6 ACL are configured on the same interface for the first time, there is a delay in filtering when the first ACL applied (in the sequence) is reattached.
- If an interface has both IPv4 ACL and IPv6 ACL configured, there is a delay in filtering when one unique ACL is removed and the other unique ACL is attached. For example, IPv4 ACL is x and IPv4 ACL is y on an interface. When x is removed and y is reattached there is a delay in filtering.
- A single IPv6 ACL attached to an interface or multiple interfaces works as a unique ACL when compared to a combination of IPv6 ACL and IPv4 ACL applied to one or more interfaces.
- The ACL attachment is rejected and may result in TCAM overflow, when you attach an IPv6 ACL where an IPv4 ACL was attached and the reverser.
- The IPv6 ACL attachment, for the first time, to an interface is rejected and may result in TCAM overflow when there is no IPv4 ACL attached.
- If a IPv6 ACL is detached from an interface, which also has IPv4 ACL attached to it may result in TCAM overflow of the IPv4 ACL and detach of both ACLs from that interface.

Information About IPv6 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 Access Control Lists

Configuring IPv6 Traffic Filtering

Creating and Configuring an IPv6 ACL for Traffic Filtering

Note IPv6 ACLs on the Cisco ASR 1000 platform do not contain implicit permit rules. The IPv6 neighbor discovery process uses the IPv6 network-layer service; therefore, to enable IPv6 neighbor discovery, you must add IPv6 ACLs to allow IPv6 neighbor discovery packets to be sent and received on an interface. In IPv4, the Address Resolution Protocol (ARP), which is equivalent to the IPv6 neighbor discovery process, uses a separate data-link-layer protocol; therefore, by default IPv4 ACLs implicitly allow ARP packets to be sent and received on an interface.

Procedure

	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** *access-list-name* **Example:** `Device(config)# ipv6 access-list inbound`	Defines an IPv6 ACL, and enters IPv6 access list configuration mode. • The *access-list name*argument specifies the name of the IPv6 ACL. IPv6 ACL names cannot contain a space or quotation mark, or begin with a numeral.
Step 4	Do one of the following: • **permit protocol** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix / prefix-length* | **any** | **host** *destination-ipv6-address*} [**operator** [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] • **deny** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport**	Specifies permit or deny conditions for an IPv6 ACL.

	Command or Action	Purpose

	Example:
	`Device(config-ipv6-acl)# permit tcp 2001:DB8:0300:0201::/32 eq telnet any`
	Example:
	`Device(config-ipv6-acl)# deny tcp host 2001:DB8:1::1 any log-input`	

Applying the IPv6 ACL to an Interface

Procedure

	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** *type number* **Example:** `Device(config)# interface gigabitethernet 0/0/0`	Specifies the interface type and number, and enters interface configuration mode.
Step 4	**ipv6 traffic-filter** *access-list-name* {**in**| **out**} **Example:** `Device(config-if)# ipv6 traffic-filter inbound in`	Applies the specified IPv6 access list to the interface specified in the previous step.

Controlling Access to a vty

Creating an IPv6 ACL to Provide Access Class Filtering

Procedure

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode.

	Command or Action	Purpose
	Example: `Device> enable`	• Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Device# configure terminal`	Enters global configuration mode.
Step 3	**ipv6 access-list** *access-list-name* **Example:** `Device(config)# ipv6 access-list cisco`	Defines an IPv6 ACL, and enters IPv6 access list configuration mode.
Step 4	Do one of the following: • **permit protocol** {*source-ipv6-prefix/prefix-length* | **any** | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix / prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name* • **deny** *protocol* {*source-ipv6-prefix*/*prefix-length* | **any** | **host** *source-ipv6-address*} [*operator port-number*]] {*destination-ipv6-prefix/prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dest-option-type** [*doh-number* | *doh-type*]] [**dscp** *value*] [**flow-label** *value*] [**fragments**] [**log**] [**log-input**] [**mobility**] [**mobility-type** [*mh-number* | *mh-type*]] [**routing**] [**routing-type** *routing-number*] [**sequence** *value*] [**time-range** *name*] [**undetermined-transport** **Example:** `Device(config-ipv6-acl)# permit ipv6 host 2001:DB8:0:4::32 any` **Example:**	Specifies permit or deny conditions for an IPv6 ACL.

	Command or Action	Purpose
	`Device(config-ipv6-acl)# deny ipv6 host 2001:DB8:0:6::6 any`	

Applying an IPv6 ACL to the Virtual Terminal Line

Procedure

	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	**line** [**aux**| **console**| **tty**| **vty**] *line-number*[*ending-line-number*] **Example:** `Device(config)# line vty 0 4`	Identifies a specific line for configuration and enters line configuration mode. • In this example, the **vty** keyword is used to specify the virtual terminal lines for remote console access.
Step 4	**ipv6 access-class** *ipv6-access-list-name* {**in**| **out**} **Example:** `Device(config-line)# ipv6 access-class cisco in`	Filters incoming and outgoing connections to and from the device based on an IPv6 ACL.

Configuration Examples for IPv6 Access Control Lists

Example: Verifying IPv6 ACL Configuration

In this example, the **show ipv6 access-list** command is used to verify that IPv6 ACLs are configured correctly:

```
Device> show ipv6 access-list

IPv6 access list inbound
    permit tcp any any eq bgp (8 matches) sequence 10
    permit tcp any any eq telnet  (15 matches) sequence 20
    permit udp any any  sequence 30
```

```
IPv6 access list Virtual-Access2.1#427819008151 (per-user)
    permit tcp host 2001:DB8:1::32 eq bgp host 2001:DB8:2::32 eq 11000 sequence 1
    permit tcp host 2001:DB8:1::32 eq telnet host 2001:DB8:2::32 eq 11001 sequence 2
```

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 INBOUND
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
```

Example: Controlling Access to a vty

In the following example, incoming connections to the virtual terminal lines 0 to 4 are filtered based on the IPv6 access list named acl1:

```
ipv6 access-list acl1
 permit ipv6 host 2001:DB8:0:4::2/32 any
!
line vty 0 4
 ipv6 access-class acl1 in
```

Use Case or Deployment Scenarios

Creating and Applying a IPv6 Access List

```
Device(config)#ipv6 access-list test
Device(config-ipv6-acl)#deny ipv6 10:10:10:10::1/64 20:20:20:20::1/64 log-input
Device(config-ipv6-acl)#permit ipv6 any any log
Device(config-ipv6-acl)#exit
Device(config)#int gig 0/0/5
Device(config-if)#ipv6 traffic-filter test in
Device(config-if)#end
```

```
Device# show ipv6 access-list
IPv6 access list test
    deny ipv6 10:10:10:10::/64 20:20:20:20::/64 log-input (100 matches) sequence 10
    permit ipv6 any any log sequence 20
```

```
Device# show ipv6 access-list test
IPv6 access list test
    deny ipv6 10:10:10:10::/64 20:20:20:20::/64 log-input (100 matches) sequence 10
    permit ipv6 any any log sequence 20
```

```
Device#show platform hardware pp active feature acl ipv6 test
ACL Name: test
```

```
Independent TCAM Consumption: 3
Unique Combination Count: 1
Total TCAM Consumption: 3
Ingress Label: 1

Ingress Bind Point Details:

Interface: GigabitEthernet0/0/5 :
IPv4 ACL: None (Label: 0)
ACL Handle: 1
```

```
Device#show platform hardware pp active feature acl l3-acl aclhandles
Details of RSP3 L3 ACL Handles

ACL Handle  Ref Count   IPv4 ACL                  IPv4 ACL Label        IPv6 ACL
          IPv6 ACL Label
-------------------------------------------------------------------------------------------------------

 1           1          None                       0                   test
           1
```

```
Device#show platform hardware pp active feature acl l3-acl resource-summary
RSP3 L3 ACL Resource Summary

Type                  Total       Used        Free
-----------------------------------------------------------
ACL Handles           512          1          511
IPv4 ACL TCAM         1000        16          984
IPv6 ACL TCAM         128          7          121
```

Additional References

Related Documents

Related Topic	Document Title
IP access list commands	*Cisco IOS Security Command Reference*
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