Layer 3 Access Control Lists on EVCs

The ability to filter packets in a modular and scalable way is important for both network security and network management. Access Control Lists (ACLs) provide the capability to filter packets at a fine granularity. In Metro Ethernet networks, ACLs are directly applied on Ethernet virtual circuits (EVCs).

Earlier, the layer 3 ACLs were only supported on the routed ports (physical ports or BDIs). The support of layer 3 ACLs on EVCs provides the capability to filter the layer 3 packets on layer 2 bridges that support Ethernet services.

How ACL works

An ACL is a set of rules (or permit or deny statements) for identifying traffic based on criteria such as source IP address, destination IP address, and port number. ACLs are essentially used for packet filtering. A packet filter drops packets that match a deny rule and permits packets that match a permit rule. ACLs are also widely used by many modules, for example, QoS and IP routing, for traffic identification.

The following list shows the types of configurations of ACLs:

- The ingress interface can be configured with layer 3 ACL over EVC.
- The egress interface can be configured with EVC in same Bridge-Domain.

Access List Process and Rules

- The packets are filtered based on the source or destination address or the protocol against the conditions (ACEs) in the access-list.
- The incoming packet is compared to ACL entries based on the order that the entries occur in the router.
- If a packet does not match an ACE, the packet is then matched against the next ACE in the list.
- If a packet and an access list statement match, the rest of the statements in the list are skipped.
- If no conditions match an ACE, the packet is dropped.

IPv4 ACLs

- IPv4 ACLs support matching on all the same fields in IP ACLs, which include bitwise matching on IP source and destination fields, DSCP, upper layer protocol values, TCP and UDP port numbers, and TCP flags.
- IPv4 ACLs can be applied to EVCs on ingress direction.
- Both IOS numbered and named IP ACL syntax are supported.
- IPv4 ACL only apply to IPv4 packets

IPv6 ACLs

- IPv6 ACLs support matching on all IPV6 source and destination address, TCP and UDP port numbers.
- The IPv6 ACL CLI is used to configure the IPv6 ACLs.
- IPv6 ACLs only apply to IPv6 packets.

EVCs

An Ethernet virtual circuit (EVC) as defined by the Metro Ethernet Forum is a port-level point-to-point or multipoint-to-multipoint Layer 2 circuit. It is an end-to-end representation of a single instance of a Layer 2 service being offered by a provider to a customer. An EVC contains the different parameters on which the service is being offered. A service instance is the instantiation of an EVC on a specified port.

Service instances are configured under a port channel. The traffic carried by the service instance is load balanced across member links. Service instances under a port channel are grouped and each group is associated with one member link. Ingress traffic for a single EVC can arrive on any member of the bundle. All egress traffic for a service instance uses only one of the member links. Load balancing is achieved by grouping service instances and assigning them to a member link.

Ethernet virtual connection services (EVCS) uses the EVCs and service instances to provide Layer 2 switched Ethernet services. EVC status can be used by a customer edge (CE) device either to find an alternative path to the service provider network or in some cases, to fall back to a backup path over Ethernet or over another alternative service such as ATM.

Information about ACLs

This section shows the information about ACLs.

Logging

Logging is a mechanism where the entries of ACL that are matched are recorded via a logging mechanism. ACL entries that have logging enabled are sent to a logging queue, which then sends packets to the logging buffer to enable logging. The ACL counters depend on the number of packets that hit the hardware queue and not on the number of packets sent.

The first packet that triggers the ACL causes a logging message, and subsequent packets are collected over 5-minute intervals before they appear or are logged. The logging message includes the access list number, whether the packet was permitted or denied, the source IP address of the packet, and the number of packets from that source permitted or denied in the prior 5-minute interval.

IP Fragmentation

Fragmentation of IP datagrams leads to problems in matching IP access lists that match on layer 4 fields. This is because only the first fragment of the IP datagram contains the layer 4 information. Hence, only the first fragment can be used to match the layer 4 information in the ACLs. In the case of ACEs that match on layer 4 fields and permits packets, this case is circumvented by creating two entries for a single ACE, one that matches on the first fragment in the packet and matches on the L4 information and another entry that matches on the non first fragment and layer 3 fields in the ACE. The ACE entries that match on layer 4 fields and drop packets are programmed to match on the first fragment with the layer 4 fields.

Prerequisites for Layer 3 ACLs on EVCs

- Knowledge of how service instances must be configured.
- ACLs are applied in the TCAM.

Restrictions for Layer 3 ACLs on EVCs

- Layer 3 ACL on EVC is not supported on egress direction.
- Layer 3 ACL is not supported on Trunk-EFP.
- IPv4 ACL and IPv6 ACL are not supported on same EVC.
- Layer 2 ACL and layer 3 ACL are not supported on the same EVC.
- QOS and ACL are not supported on same EVC.
- Layer 3 ACL on EVC on port-channel having member links on different ASICs is not supported.
- The maximum number of ACE entries supported is 512.
- When layer 3 ACL is configured on EVC and the corresponding BDI, ACL configured on BDI takes priority.
- Layer 3 ACL on EVC is not supported when QoS policy is attached on the interface.

Configuring Layer 3 ACL on EVCs

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list standard** *name*
4. **permit** / **deny** {*source* [*source-wildcard*] | **any**} [**log**]
5. **access-list** *access-list-number* **permit** / **deny** {*source* [*source-wildcard*] | **any**} [**log**]
6. **ip access-list extended** *name*
7. **permit** / **deny** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
8. **access-list** *access-list number* **permit** *protocol* {**source** [*source-wildcard*] | *any*} {*destination* [*destination-wildcard*] | *any*} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]

DETAILED STEPS

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

	Command or Action	Purpose
Step 3	**ip access-list standard** *name*	Defines a standard IP access list using a name and enters standard named access list configuration mode.
Step 4	**permit** / **deny** {*source* [*source-wildcard*] | **any**} [**log**]	Permits or denies the specified source based on a source address and wildcard mask. • Every access list needs at least one permit statement; it need not be the first entry. • If the source-wildcard is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally use the keyword **any** as a substitute for the source source-wildcard to specify the source and source wildcard of 0.0.0.0 255.255.255.255. **Note** There is another way of configuration between the steps 4 and 5.
Step 5	**access-list** *access-list-number* **permit** / **deny** {*source* [*source-wildcard*] | **any**} [**log**]	Permits or denies the specified source based on a source address and wildcard mask. • Every access list needs at least one permit statement; it need not be the first entry. • Standard IP access lists are numbered 1 to 99 or 1300 to 1999. • If the source-wildcard is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally use the keyword **any** as a substitute for the source source-wildcard to specify the source and source wildcard of 0.0.0.0 255.255.255.255.
Step 6	**ip access-list extended** *name*	Defines an extended IP access list using a name and enters extended named access list configuration mode.
Step 7	**permit** / **deny** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]	Permits or denies any packet that matches all of the conditions specified in the statement. • Every access list needs at least one permit statement. • If the source-wildcard or destination-wildcard is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source or destination address, respectively. • Optionally use the keyword **any** as a substitute for the source source-wildcard or destination destination-wildcard to specify the address and wildcard of 0.0.0.0 255.255.255.255. • Optionally use the keyword **host source** to indicate a source and source wildcard of source 0.0.0.0 or the abbreviation host destination to indicate a destination and destination wildcard of destination 0.0.0.0. • Use the **log-input** keyword to include input interface, source MAC address, or virtual circuit in the logging output.

	Command or Action	Purpose
Step 8	**access-list** *access-list number* **permit** *protocol* {**source** [*source-wildcard*] | *any*} {*destination* [*destination-wildcard*] | *any*} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]	Permits any packet that matches all of the conditions specified in the statement. • Every access list needs at least one permit statement; it need not be the first entry. • Extended IP access lists are numbered 100 to 199 or 2000 to 2699. • If the *source-wildcard* or *destination-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source or destination address, respectively. • Optionally use the keyword **any** as a substitute for the *source source-wildcard* or *destination destination-wildcard* to specify the address and wildcard of 0.0.0.0 255.255.255.255. • TCP and other protocols have additional syntax available. See the access-list command in the command reference for complete syntax.

Applying IPv4 ACL on EVC

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** <*interface-number*>
4. **service instance** *id* **ethernet**
5. **ip access-group** {*access-list-number* | *access-list-name*} **in**
6. **end**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters global configuration mode.
Step 3	**interface** <*interface-number*>	Enters the interface configuration mode for the interface connected to the service-provider network. You can enter a physical interface or an EtherChannel port channel.

	Command or Action	Purpose
Step 4	**service instance** *id* **ethernet**	Configures an Ethernet service instance on the interface and enters Ethernet service configuration mode. • The Ethernet service instance identifier is a per-interface service identifier and does not map to a VLAN.
Step 5	**ip access-group** {*access-list-number* | *access-list-name*} **in**	Applies the specified access list to the inbound interface. • To filter source addresses, apply the access list to the inbound interface.
Step 6	**end**	Returns to privileged EXEC mode.

Creating a Standard Access List to Filter on Source Address

If you want to filter on source address only, a standard access list is simple and sufficient. There are two alternative types of standard access list: named and numbered. Named access lists allow you to identify your access lists with a more intuitive name rather than a number, and they also support more features than numbered access lists.

Creating a Named Access List to Filter on Source Address

Use a standard, named access list if you need to filter on source address only. This task illustrates one permit statement and one deny statement, but the actual statements you use and their order depend on what you want to filter or allow. Define your permit and deny statements in the order that achieves your filtering goals.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list standard** *iname*
4. **remark** *remark*
5. **deny** {*source* [*source-wildcard*] |**any**} [**log**]
6. **remark** *remark*
7. **permit** {*source* [*source-wildcard*] |**any**} [**log**]
8. Repeat some combination of Steps 4 through 7 until you have specified the sources on which you want to base your access list.
9. **end**
10. **show ip access-list**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters global configuration mode.
Step 3	**ip access-list standard** *iname*	Defines a standard IP access list using a name and enters standard named access list configuration mode.
Step 4	**remark** *remark*	(Optional) Adds a user-friendly comment about an access list entry. • A remark can precede or follow an access list entry.
Step 5	**deny** {*source* [*source-wildcard*] |**any**} [**log**]	(Optional) Denies the specified source based on a source address and wildcard mask. • If the *source-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally, use the keyword any as a substitute for the *source source-wildcard* to specify the source and source wildcard of 0.0.0.0 255.255.255.255
Step 6	**remark** *remark*	(Optional) Adds a user-friendly comment about an access list entry. • This remark reminds the network administrator that the subsequent entry allows the Tester's host access to the interface.
Step 7	**permit** {*source* [*source-wildcard*] |**any**} [**log**]	Permits the specified source based on a source address and wildcard mask. • Optionally, use the keyword any as a substitute for the *source source-wildcard* to specify the source and source wildcard of 0.0.0.0 255.255.255.255 • If the *source-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address.
Step 8	Repeat some combination of Steps 4 through 7 until you have specified the sources on which you want to base your access list.	Remember that all sources not specifically permitted are denied by an implicit deny statement at the end of the access list.
Step 9	**end**	Returns to privileged EXEC mode.
Step 10	**show ip access-list**	(Optional) Displays the contents of all current IP access lists.

Creating a Numbered Access List to Filter on Source Address

Configure a standard, numbered access list if you need to filter on source address only and you prefer not to use a named access list.

IP standard access lists are numbered 1 to 99 or 1300 to 1999. This task illustrates one permit statement and one deny statement, but the actual statements you use and their order depend on what you want to filter or allow. Define your **permit** and **deny** statements in the order that achieves your filtering goals.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **permit** {*source* [*source-wildcard*] | **any**} [**log**]
4. **access-list** *access-list-number* **deny** {*source* [*source-wildcard*] | **any**} [**log**]
5. Repeat some combination of Steps 3 through 6 until you have specified the sources on which you want to base your access list.
6. **end**
7. **show ip access-list**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters global configuration mode.
Step 3	**access-list** *access-list-number* **permit** {*source* [*source-wildcard*] | **any**} [**log**]	Permits the specified source based on a source address and wildcard mask. • Every access list needs at least one permit statement; it need not be the first entry. • Standard IP access lists are numbered 1 to 99 or 1300 to 1999. • If the source-wildcard is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address. • Optionally, use the keyword any as a substitute for the source source-wildcard to specify the source and source wildcard of 0.0.0.0 255.255.255.255.
Step 4	**access-list** *access-list-number* **deny** {*source* [*source-wildcard*] | **any**} [**log**]	Denies the specified source based on a source address and wildcard mask. • If the *source-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address.

	Command or Action	Purpose
		• Optionally, use the abbreviation any as a substitute for the *source source-wildcard* to specify the source and source wildcard of 0.0.0.0 255.255.255.255. • If the source-wildcard is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source address.
Step 5	Repeat some combination of Steps 3 through 6 until you have specified the sources on which you want to base your access list.	Remember that all sources not specifically permitted are denied by an implicit deny statement at the end of the access list.
Step 6	**end**	Exits global configuration mode and enters privileged EXEC mode.
Step 7	**show ip access-list**	(Optional) Displays the contents of all current IP access lists.

Creating Numbered Layer 3 ACL

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list standard***name*
4. **standard ip access-list** *number*
5. **standard ip access-list extended** *name*
6. **access-list** *name*

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters global configuration mode.
Step 3	**ip access-list standard***name*	Defines a standard IP access list using a name and enters standard named access list configuration mode.
Step 4	**standard ip access-list** *number*	Applies the specified access list to the inbound interface.

	Command or Action	Purpose
Step 5	**standard ip access-list extended** *name*	Defines an extended IP access list using a name and enters extended named access list configuration mode
Step 6	**access-list** *name*	Applies the specified access list to the inbound interface

Creating an Extended Access List

If you want to filter on anything other than source address, you need to create an extended access list. There are two alternative types of extended access list: named and numbered. Named access lists allow you to identify your access lists with a more intuitive name rather than a number, and they also support more features.

For details on how to filter something other than source or destination address, see the syntax descriptions in the command reference documentation.

Creating a Named Extended Access List

Create a named extended access list if you want to filter on source and destination address, or a combination of addresses and other IP fields.

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ip access-list extended** *name*
4. **remark** *remark*
5. **deny** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
6. **remark** *remark*
7. **permit** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
8. Repeat some combination of Steps 4 through 7 until you have specified the fields and values on which you want to base your access list.
9. **end**
10. **show ip access-list**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters global configuration mode.
Step 3	**ip access-list extended** *name*	Defines an extended IP access list using a name and enters extended named access list configuration mode.
Step 4	**remark** *remark*	(Optional) Adds a user-friendly comment about an access list entry. • A remark can precede or follow an access list entry.
Step 5	**deny** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]	(Optional) Denies any packet that matches all of the conditions specified in the statement. • If the *source-wildcard* or *destination-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source or destination address, respectively. • Optionally use the keyword any as a substitute for the *source source-wildcard* or *destination destination-wildcard* to specify the address and wildcard of 0.0.0.0 255.255.255.255. • Optionally use the keyword host source to indicate a source and source wildcard of source 0.0.0.0 or the abbreviation host destinationto indicate a destination and destination wildcard of destination 0.0.0.0.
Step 6	**remark** *remark*	(Optional) Adds a user-friendly comment about an access list entry. • A remark can precede or follow an access list entry.
Step 7	**permit** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*] [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]	Permits any packet that matches all of the conditions specified in the statement. • Every access list needs at least one permit statement. • If the *source-wildcard* or *destination-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source or destination address, respectively. • Optionally, use the keyword any as a substitute for the *source source-wildcard* or *destination destination-wildcard* to specify the address and wildcard of 0.0.0.0 255.255.255.255. • The **log-input** keyword can be configured, but it is not supported, and will not work as expected

	Command or Action	Purpose
Step 8	Repeat some combination of Steps 4 through 7 until you have specified the fields and values on which you want to base your access list.	Remember that all sources not specifically permitted are denied by an implicit **deny** statement at the end of the access list.
Step 9	**end**	Exits global configuration mode and enters privileged EXEC mode.
Step 10	**show ip access-list**	(Optional) Displays the contents of all current IP access lists.

Creating a Numbered Extended Access List

Create a numbered extended access list if you want to filter on source and destination address, or a combination of addresses and other IP fields, and you prefer not to use a name. Extended IP access lists are numbered 100 to 199 or 2000 to 2699

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **access-list** *access-list-number* **remark** *remark*
4. **access-list** *access-list-number* **permit** *protocol* {*source* [*source-wildcard*] | **any**} {*destination* [*destination-wildcard*] | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
5. **access-list** *access-list-number* **remark** *remark*
6. **access-list** *access-list-number* **deny** *protocol* {*source* [*source-wildcard*] | **any**} {*destination* [*destination-wildcard*] | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]
7. Repeat some combination of Steps 3 through 6 until you have specified the fields and values on which you want to base your access list.
8. **end**
9. **show ip access-list**

DETAILED STEPS

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

	Command or Action	Purpose
Step 3	**access-list** *access-list-number* **remark** *remark*	(Optional) Adds a user-friendly comment about an access list entry. • A remark of up to 100 characters can precede or follow an access list entry.
Step 4	**access-list** *access-list-number* **permit** *protocol* {*source* [*source-wildcard*] | **any**} {*destination* [*destination-wildcard*] | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]	Permits any packet that matches all of the conditions specified in the statement. • Every access list needs at least one permit statement; it need not be the first entry. • Extended IP access lists are numbered 100 to 199 or 2000 to 2699. • If the *source-wildcard* or *destination-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source or destination address, respectively. • Optionally, use the keyword any as a substitute for the *source source-wildcard* or *destination destination-wildcard* to specify the address and wildcard of 0.0.0.0 255.255.255.255. • TCP and other protocols have additional syntax available. See the **access-list** command in the command reference for complete syntax
Step 5	**access-list** *access-list-number* **remark** *remark*	(Optional) Adds a user-friendly comment about an access list entry. • A remark of up to 100 characters can precede or follow an access list entry.
Step 6	**access-list** *access-list-number* **deny** *protocol* {*source* [*source-wildcard*] | **any**} {*destination* [*destination-wildcard*] | **any**} [**precedence** *precedence*] [**tos** *tos*] [**established**] [**log** | **log-input**] [**time-range** *time-range-name*] [**fragments**]	Denies any packet that matches all of the conditions specified in the statement. • If the *source-wildcard* or *destination-wildcard* is omitted, a wildcard mask of 0.0.0.0 is assumed, meaning match on all bits of the source or destination address, respectively. • Optionally, use the keyword any as a substitute for the *source source-wildcard* or *destination destination-wildcard* to specify the address and wildcard of 0.0.0.0 255.255.255.255.
Step 7	Repeat some combination of Steps 3 through 6 until you have specified the fields and values on which you want to base your access list.	Remember that all sources not specifically permitted are denied by an implicit **deny** statement at the end of the access list.
Step 8	**end**	Exits global configuration mode and enters privileged EXEC mode.
Step 9	**show ip access-list**	(Optional) Displays the contents of all current IP access lists.

Creating IPv6 ACL

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **ipv6 access-list** *access-list-name*
4. **deny** | **permit** *protocol* {*source-ipv6-prefix* | *prefix-length* | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix* | *prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**sequence** *value*] [**time-range** *name*]

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable**	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal**	Enters global configuration mode.
Step 3	**ipv6 access-list** *access-list-name*	Defines an IPv6 ACL, and enters IPv6 access list configuration mode. • The access-list nameargument specifies the name of the IPv6 ACL. IPv6 ACL names cannot contain a space or quotation mark, or begin with a numeral.
Step 4	**deny** | **permit** *protocol* {*source-ipv6-prefix* | *prefix-length* | **host** *source-ipv6-address*} [*operator* [*port-number*]] {*destination-ipv6-prefix* | *prefix-length* | **any** | **host** *destination-ipv6-address*} [*operator* [*port-number*]] [**dscp** *value*] [**fragments**] [**log**] [**log-input**] [**sequence** *value*] [**time-range** *name*]	Enter deny or permit to specify whether to deny or permit the packet if conditions are matched. These are the conditions: • For protocol, enter the name or number of an Internet protocol: ahp, esp, icmp, ipv6, pcp, sctp, tcp, or udp, or an integer in the range 0 to 255 representing an IPv6 protocol number. For additional specific parameters for ICMP, TCP, and UDP, see Steps 3b through 3d. • The *source-ipv6-prefix* | *prefix-length* or *destination-ipv6-prefix* | *prefix-length* is the source or destination IPv6 network or class of networks for which to set deny or permit conditions, specified in hexadecimal and using 16-bit values between colons. • Enter any as an abbreviation for the IPv6 prefix ::/0. • For **host** *source-ipv6-address* or *destination-ipv6-address*, enter the source or destination IPv6 host address for which to set deny or permit conditions, specified in hexadecimal using 16-bit values between colons. • (Optional) For operator, specify an operand that compares the source or destination ports of the specified protocol. Operands are lt (less than), gt (greater than), eq (equal), neq (not equal), and range. If the operator follows the source-ipv6-prefix/prefix-length argument, it must match the source port. If the

	Command or Action	Purpose
		operator follows the destination-ipv6- prefix/prefix-length argument, it must match the destination port. • (Optional) The port-number is a decimal number from 0 to 65535 or the name of a TCP or UDP port for filtering TCP or UDP, respectively. • (Optional) Enter dscp value to match a differentiated services code point value against the traffic class value in the Traffic Class field of each IPv6 packet header. The acceptable range is from 0 to 63. • (Optional) Enter fragments to check noninitial fragments. This keyword is visible only if the protocol is ipv6. • (Optional) Enter log to cause an logging message to be sent to the console about the packet that matches the entry. Enter log-input to include the input interface in the log entry. Logging is supported only for router ACLs. • (Optional) Enter sequence value to specify the sequence number for the access list statement. The acceptable range is from 1 to 4294967295. • (Optional) Enter time-range name to specify a time range for the statement.

Applying IPv6 ACL on EVC

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** <*interface-number*>
4. **service instance** *id* **ethernet**
5. **ipv6 traffic-filter** *access-list-name in*

DETAILED STEPS

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

	Command or Action	Purpose
Step 3	**interface** <*interface-number*>	Enters the interface configuration mode for the interface connected to the service-provider network. You can enter a physical interface or an EtherChannel port channel.
Step 4	**service instance** *id* **ethernet**	Configures an Ethernet service instance on the interface and enters Ethernet service configuration mode. • The Ethernet service instance identifier is a per-interface service identifier and does not map to a VLAN.
Step 5	**ipv6 traffic-filter** *access-list-name in*	Defines an IPv6 ACL, and enters IPv6 access list configuration mode.

Configuration Examples for IPv4 ACLs on EVC

```
Building configuration...

Current configuration : 207 bytes
!
interface GigabitEthernet0/0/4
 no ip address
 media-type auto-select
 negotiation auto
 service instance 1 ethernet
  encapsulation dot1q 1
  rewrite ingress tag pop 1 symmetric
  bridge-domain 1
 !
end
Building configuration...

Current configuration : 111 bytes
!
interface BDI1
 ip address 20.0.0.1 255.255.255.0
end

Node1#sh ip access
Node1#sh ip access-list
IPv4 access list ipv4_acl
    permit udp host 20::4 eq 10 any eq 30 log sequence 10
Node1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Node1(config)#int g 0/0/4
Node1(config-if)#int BDI 1
Node1(config-if)#ip access-group ipv4_acl in
Node1(config-if-srv)#end
Node1#sh access-l
Oct 19 12:48:41.580 IST: %SYS-5-CONFIG_I: Configured from console by consolei
Node1#sh access-lists
Extended IP access list ip_acl_25
    10 permit udp host 20.0.0.4 eq 10 any eq 30 log
IPv4 access list ipv4_acl
    permit udp host 20::4 eq 10 any eq 30 log sequence 10
Node1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Node1(config)#int g 0/0/4
Node1(config-if)#int BDI 1
```

```
Node1(config-if)#ip access-group ipv4_acl in
Node1(config-if)#
Oct 19 12:49:26.330 IST: %IPV4_ACL-6-ACCESSLOGP: list ipv4_acl/10 permitted udp 20::4(10)
-> 30::2(30), 1 packet
Node1(config-if)#
Node1(config-if)#do sh access-li
Extended IP access list ip_acl_25
    10 permit udp host 20.0.0.4 eq 10 any eq 30 log
IPv4 access list ipv4_acl
    permit udp host 20::4 eq 10 any eq 30 log (5705 matches) sequence 10
Node1(config-if)#
```

Configuration Examples for IPv6 ACLs on EVC

```
Building configuration...
Current configuration : 207 bytes
!
interface GigabitEthernet0/0/4
no ip address
media-type auto-select
negotiation auto
service instance 1 ethernet
encapsulation dot1q 1
rewrite ingress tag pop 1 symmetric
bridge-domain 1
!
end
Building configuration...
Current configuration : 111 bytes
!
interface BDI1
ip address 20.0.0.1 255.255.255.0
ip ospf 1 area 0
ipv6 address 20::1/64
ipv6 enable
end
Node1#sh ipv6 access
Node1#sh ipv6 access-list
IPv6 access list ipv6_acl
permit udp host 20::4 eq 10 any eq 30 log sequence 10
Node1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Node1(config)#int ser
Node1(config)#int g 0/0/4
Node1(config-if)#ser in 1 eth
Node1(config-if-srv)#ipv6 traff
Node1(config-if-srv)#ipv6 traffic-filter ipv6_acl in
Node1(config-if-srv)#end
Node1#sh access-l
Oct 19 12:48:41.580 IST: %SYS-5-CONFIG_I: Configured from console by consolei
Node1#sh access-lists
Extended IP access list ip_acl_25
10 permit udp host 20.0.0.4 eq 10 any eq 30 log
IPv6 access list ipv6_acl
permit udp host 20::4 eq 10 any eq 30 log sequence 10
Node1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Node1(config)#int g 0/0/4
Node1(config-if)#int BDI 1
Node1(config-if)#ipv6 tra
Node1(config-if)#ipv6 traffic-filter ipv6_acl in
Node1(config-if)#
Oct 19 12:49:26.330 IST: %IPV6_ACL-6-ACCESSLOGP: list ipv6_acl/10 permitted udp 20::4(10)
-> 30::2(30), 1 packet
Node1(config-if)#
Node1(config-if)#do sh access-li
Extended IP access list ip_acl_25
10 permit udp host 20.0.0.4 eq 10 any eq 30 log
IPv6 access list ipv6_acl
```

```
permit udp host 20::4 eq 10 any eq 30 log (5705 matches) sequence 10
Node1(config-if)#
```

Verification of Layer 3 ACLs on EVCs

Use the **show running interface** command to verify the Layer 3 ACLs attached on EVCs.

```
Node2#sh run int g 0/0/4
Building configuration...

Current configuration : 237 bytes
!
interface GigabitEthernet0/0/4
no ip address
media-type auto-select
negotiation auto
service instance 1 ethernet
encapsulation dot1q 1
rewrite ingress tag pop 1 symmetric
ip access-group ipv4_acl in
bridge-domain 1
!
end
```

Troubleshooting Guidelines and Commands

Execute the following commands to check the ACL counters:

- **show platform hardware pp active acl ipv4 name < acl -name > stats**
- **show platform hardware pp active acl ipv6 name < acl -name > stats**

Execute the following commands to check TCAM entries:

- **show platform hardware pp active tcam utilization acl detail < asic -id >**
- **show platform hardware pp active tcam utilization ipv6-acl detail <asic-id>**

Execute the following command to check the layer 3 ACLs attached on EVCs:

- **show running interface <id>**

Execute the following commands for ACL debugging:

- **show platform hardware pp ac nqatm <asic-id> acl all**
- **show platform hardware pp ac nqatm <asic-id> ipv6-acl all**