# Security


- aaa accounting, on page 5
- aaa accounting dot1x, on page 8
- aaa accounting identity, on page 10
- aaa authentication dot1x, on page 12
- aaa authorization, on page 14
- aaa common-criteria policy, on page 18
- aaa new-model, on page 20
- access-session mac-move deny, on page 22
- access-session host-mode multi-host, on page 24
- action, on page 26
- api-key (Parameter Map), on page 27
- authentication host-mode, on page 28
- authentication logging verbose, on page 30
- authentication mac-move permit, on page 31
- authentication priority, on page 33
- authentication timer reauthenticate, on page 35
- authentication violation, on page 37
- cisp enable, on page 39
- clear aaa cache group, on page 40
- clear device-tracking database, on page 41
- clear errdisable interface vlan, on page 45
- clear fqdn, on page 46
- clear mac address-table, on page 47
- confidentiality-offset, on page 49
- crypto pki trustpool import, on page 50
- debug aaa cache group, on page 53
- debug aaa dead-criteria transaction, on page 54
- debug umbrella, on page 56
- delay-protection, on page 57
- deny (MAC access-list configuration), on page 58
- device-role (IPv6 snooping), on page 61
- device-role (IPv6 nd inspection), on page 62
- device-tracking (interface config), on page 63

- device-tracking (VLAN config), on page 66
- device-tracking binding, on page 69
- device-tracking logging, on page 89
- device-tracking policy, on page 93
- device-tracking tracking, on page 106
- device-tracking upgrade-cli, on page 110
- device sensor filter list mdns, on page 113
- device sensor filter spec, on page 114
- dnscrypt (Parameter Map), on page 116
- dot1x authenticator eap profile, on page 117
- dot1x critical (global configuration), on page 118
- dot1x logging verbose, on page 119
- dot1x max-start, on page 120
- dot1x pae, on page 121
- dot1x supplicant controlled transient, on page 122
- dot1x supplicant force-multicast, on page 123
- dot1x test eapol-capable, on page 124
- dot1x test timeout, on page 125
- dot1x timeout, on page 126
- dscp, on page 128
- dtls, on page 129
- enable password, on page 131
- enable secret, on page 134
- epm access-control open, on page 137
- fqdn ttl-timeout factor, on page 138
- include-icv-indicator, on page 139
- ip access-list, on page 140
- ip access-list role-based, on page 143
- ip admission, on page 144
- ip admission name, on page 145
- ip dhcp restrict-next-hop, on page 147
- ip dhcp snooping database, on page 148
- ip dhcp snooping information option format remote-id, on page 150
- ip dhcp snooping verify no-relay-agent-address, on page 151
- ip http access-class, on page 152
- ip radius source-interface, on page 154
- ip source binding, on page 156
- ip ssh pubkey-chain, on page 157
- ip ssh source-interface, on page 158
- ip verify source, on page 159
- ipv6 access-list, on page 160
- ipv6 snooping policy, on page 162
- key chain macsec, on page 163
- key config-key password-encrypt, on page 164
- key-hash, on page 166
- key-server, on page 168

- limit address-count, on page 169
- local-domain (Parameter Map), on page 170
- mab logging verbose, on page 171
- mab request format attribute 32, on page 172
- macsec-cipher-suite, on page 174
- macsec network-link, on page 176
- match (access-map configuration), on page 177
- mka pre-shared-key, on page 179
- mka suppress syslogs sak-rekey, on page 180
- orgid (Parameter Map), on page 181
- parameter-map type regex, on page 182
- parameter-map type umbrella global, on page 185
- password encryption aes, on page 186
- pattern (Parameter Map), on page 188
- permit (MAC access-list configuration), on page 190
- protocol (IPv6 snooping), on page 193
- radius server, on page 194
- radius-server dscp, on page 196
- radius-server dead-criteria, on page 197
- radius-server deadtime, on page 199
- radius-server directed-request, on page 201
- radius-server domain-stripping, on page 203
- sak-rekey, on page 207
- secret (Parameter Map), on page 208
- security level (IPv6 snooping), on page 209
- security passthru, on page 210
- send-secure-announcements, on page 211
- server-private (RADIUS), on page 212
- server-private (TACACS+), on page 214
- show aaa cache group, on page 216
- show aaa clients, on page 218
- show aaa command handler, on page 219
- show aaa common-criteria policy, on page 220
- show aaa dead-criteria, on page 222
- **show aaa local**, on page 225
- show aaa servers, on page 227
- show aaa sessions, on page 228
- show access-lists fqdn statistics, on page 229
- show access-session, on page 230
- show authentication brief, on page 235
- show authentication history, on page 238
- show authentication sessions, on page 239
- show cisp, on page 242
- show device-tracking capture-policy, on page 244
- show device-tracking counters, on page 246
- show device-tracking database, on page 248

- show device-tracking events, on page 253
- show device-tracking features, on page 255
- show device-tracking messages, on page 256
- show device-tracking policies, on page 257
- show device-tracking policy, on page 258
- show dot1x, on page 259
- show eap pac peer, on page 261
- show fqdn, on page 262
- show ip access-lists, on page 264
- show ip dhcp snooping statistics, on page 267
- show platform software dns-umbrella statistics, on page 270
- show platform software umbrella switch F0, on page 271
- show radius server-group, on page 273
- show storm-control, on page 275
- show tech-support acl, on page 277
- show tech-support identity, on page 281
- show umbrella, on page 290
- show vlan access-map, on page 292
- show vlan filter, on page 293
- show vlan group, on page 294
- ssci-based-on-sci, on page 295
- storm-control, on page 296
- switchport port-security aging, on page 299
- switchport port-security mac-address, on page 301
- switchport port-security maximum, on page 304
- switchport port-security violation, on page 306
- tacacs server, on page 308
- tls, on page 309
- tlv (mDNS), on page 311
- token (Parameter Map), on page 312
- tracking (IPv6 snooping), on page 313
- trusted-port, on page 315
- umbrella, on page 316
- use-updated-eth-header, on page 317
- username, on page 318
- vlan access-map, on page 323
- vlan dot1Q tag native, on page 325
- vlan filter, on page 326
- vlan group, on page 327

# aaa accounting

To enable authentication, authorization, and accounting (AAA) accounting of requested services for billing or security purposes when you use RADIUS or TACACS+, use the **aaa accounting** command in global configuration mode. To disable AAA accounting, use the **no** form of this command.

**aaa accounting** {**auth-proxy** | **system** | **network** | **exec** | **connections** | **commands** *level*} {**default** | *list-name*} {**start-stop** | **stop-only** | **none**} [**broadcast**] **group** *group-name*
**no aaa accounting** {**auth-proxy** | **system** | **network** | **exec** | **connections** | **commands** *level*} {**default** | *list-name*} {**start-stop** | **stop-only** | **none**} [**broadcast**] **group** *group-name*

## Syntax Description

| | |
|---|---|
| **auth-proxy** | Provides information about all authenticated-proxy user events. |
| **system** | Performs accounting for all system-level events not associated with users, such as reloads. |
| **network** | Runs accounting for all network-related service requests. |
| **exec** | Runs accounting for EXEC shell session. This keyword might return user profile information such as what is generated by the **autocommand** command. |
| **connection** | Provides information about all outbound connections made from the network access server. |
| **commands** *level* | Runs accounting for all commands at the specified privilege level. Valid privilege level entries are integers from 0 through 15. |
| **default** | Uses the listed accounting methods that follow this argument as the default list of methods for accounting services. |
| *list-name* | Character string used to name the list of at least one of the accounting methods described in |
| **start-stop** | Sends a "start" accounting notice at the beginning of a process and a "stop" accounting notice at the end of a process. The "start" accounting record is sent in the background. The requested user process begins regardless of whether the "start" accounting notice was received by the accounting server. |
| **stop-only** | Sends a "stop" accounting notice at the end of the requested user process. |
| **none** | Disables accounting services on this line or interface. |
| **broadcast** | (Optional) Enables sending accounting records to multiple AAA servers. Simultaneously sends accounting records to the first server in each group. If the first server is unavailable, fail over occurs using the backup servers defined within that group. |
| *group groupname* | At least one of the keywords described in the AAA Accounting Methods table. |

## Command Default

AAA accounting is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **aaa accounting** command to enable accounting and to create named method lists defining specific accounting methods on a per-line or per-interface basis.

***Table 1: AAA Accounting Methods***

| Keyword | Description |
|---|---|
| **group radius** | Uses the list of all RADIUS servers for authentication as defined by the **aaa group server radius** command. |
| **group tacacs+** | Uses the list of all TACACS+ servers for authentication as defined by the **aaa group server tacacs+** command. |
| **group** *group-name* | Uses a subset of RADIUS or TACACS+ servers for accounting as defined by the server group group-name. |

In AAA Accounting Methods table, the **group radius** and **group tacacs+** methods refer to a set of previously defined RADIUS or TACACS+ servers. Use the **radius server** and **tacacs server** commands to configure the host servers. Use the **aaa group server radius** and **aaa group server tacacs+** commands to create a named group of servers.

Cisco IOS XE software supports the following two methods of accounting:

- RADIUS—The network access server reports user activity to the RADIUS security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server.
- TACACS+—The network access server reports user activity to the TACACS+ security server in the form of accounting records. Each accounting record contains accounting attribute-value (AV) pairs and is stored on the security server.

Method lists for accounting define the way accounting will be performed. Named accounting method lists enable you to designate a particular security protocol to be used on specific lines or interfaces for particular types of accounting services. Create a list by entering the *list-name* and the *method* , where *list-name* is any character string used to name this list (excluding the names of methods, such as radius or tacacs+) and *method* identifies the methods to be tried in sequence as given.

If the **aaa accounting** command for a particular accounting type is issued without a named method list specified, the default method list is automatically applied to all interfaces or lines (where this accounting type applies) except those that have a named method list explicitly defined. (A defined method list overrides the default method list.) If no default method list is defined, then no accounting takes place.

**Note** System accounting does not use named accounting lists; you can only define the default list for system accounting.

For minimal accounting, include the **stop-only** keyword to send a stop record accounting notice at the end of the requested user process. For more accounting, you can include the **start-stop** keyword, so that RADIUS or TACACS+ sends a start accounting notice at the beginning of the requested process and a stop accounting

notice at the end of the process. Accounting is stored only on the RADIUS or TACACS+ server. The none keyword disables accounting services for the specified line or interface.

When AAA accounting is activated, the network access server monitors either RADIUS accounting attributes or TACACS+ AV pairs pertinent to the connection, depending on the security method you have implemented. The network access server reports these attributes as accounting records, which are then stored in an accounting log on the security server.

**Note** This command cannot be used with TACACS or extended TACACS.

This example defines a default commands accounting method list, where accounting services are provided by a TACACS+ security server, set for privilege level 15 commands with a stop-only restriction:

```
Device> enable
Device# configure terminal
Device(config)# aaa accounting commands 15 default stop-only group TACACS+
Device(config)# exit
```

This example defines a default auth-proxy accounting method list, where accounting services are provided by a TACACS+ security server with a stop-only restriction. The **aaa accounting** commands activates authentication proxy accounting.

```
Device> enable
Device# configure terminal
Device(config)# aaa new model
Device(config)# aaa authentication login default group TACACS+
Device(config)# aaa authorization auth-proxy default group TACACS+
Device(config)# aaa accounting auth-proxy default start-stop group TACACS+
Device(config)# exit
```

# aaa accounting dot1x

To enable authentication, authorization, and accounting (AAA) accounting and to create method lists defining specific accounting methods on a per-line or per-interface basis for IEEE 802.1x sessions, use the **aaa accounting dot1x**command in global configuration mode. To disable IEEE 802.1x accounting, use the **no** form of this command.

**aaa accounting dot1x** { *name* | **default** } **start-stop** { **broadcast group** { *name* | **radius** } [ **group** { *name* | **radius** } **. . .** ] | **group** { *name* | **radius** } [ **group** { *name* | **radius** } **. . .** ] }
**no aaa accounting dot1x** {*name* | **default** }

## Syntax Description

| | |
|---|---|
| *name* | Name of a server group. This is optional when you enter it after the **broadcast group** and **group** keywords. |
| **default** | Specifies the accounting methods that follow as the default list for accounting services. |
| **start-stop** | Sends a start accounting notice at the beginning of a process and a stop accounting notice at the end of a process. The start accounting record is sent in the background. The requested user process begins regardless of whether or not the start accounting notice was received by the accounting server. |
| **broadcast** | Enables accounting records to be sent to multiple AAA servers and sends accounting records to the first server in each group. If the first server is unavailable, the device uses the list of backup servers to identify the first server. |
| **group** | Specifies the server group to be used for accounting services. These are valid server group names:<br>• *name* — Name of a server group.<br>• **radius** — Lists of all RADIUS hosts.<br>• **tacacs+** — Lists of all TACACS+ hosts.<br>The **group** keyword is optional when you enter it after the **broadcast group** and **group** keywords. You can enter more than optional **group** keyword. |
| **radius** | (Optional) Enables RADIUS accounting. |

## Command Default

AAA accounting is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

This command requires access to a RADIUS server.

We recommend that you enter the **dot1x reauthentication** interface configuration command before configuring IEEE 802.1x RADIUS accounting on an interface.

This example shows how to configure IEEE 802.1x accounting:

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa accounting dot1x default start-stop group radius
Device(config)# exit
```

# aaa accounting identity

To enable authentication, authorization, and accounting (AAA) accounting for IEEE 802.1x, MAC authentication bypass (MAB), and web authentication sessions, use the **aaa accounting identity** command in global configuration mode. To disable IEEE 802.1x accounting, use the **no** form of this command.

**aaa accounting identity** {*name* | **default**} **start-stop** {**broadcast group** {*name* | **radius** | **tacacs+**} [**group** {*name* | **radius** | **tacacs+**} **...** ] | **group** {*name* | **radius** | **tacacs+**} [**group** {*name* | **radius** | **tacacs+**}**...** ]}
**no aaa accounting identity** {*name* | **default**}

## Syntax Description

| | |
|---|---|
| *name* | Name of a server group. This is optional when you enter it after the **broadcast group** and **group** keywords. |
| **default** | Uses the accounting methods that follow as the default list for accounting services. |
| **start-stop** | Sends a start accounting notice at the beginning of a process and a stop accounting notice at the end of a process. The start accounting record is sent in the background. The requested-user process begins regardless of whether or not the start accounting notice was received by the accounting server. |
| **broadcast** | Enables accounting records to be sent to multiple AAA servers and send accounting records to the first server in each group. If the first server is unavailable, the switch uses the list of backup servers to identify the first server. |
| **group** | Specifies the server group to be used for accounting services. These are valid server group names:<br>• *name* — Name of a server group.<br>• **radius** — Lists of all RADIUS hosts.<br>• **tacacs+** — Lists of all TACACS+ hosts.<br>The **group** keyword is optional when you enter it after the **broadcast group** and **group** keywords. You can enter more than optional **group** keyword. |
| **radius** | (Optional) Enables RADIUS authorization. |
| **tacacs+** | (Optional) Enables TACACS+ accounting. |

## Command Default

AAA accounting is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

To enable AAA accounting identity, you need to enable policy mode. To enable policy mode, enter the **authentication display new-style** command in privileged EXEC mode.

This example shows how to configure IEEE 802.1x accounting identity:

```
Device# authentication display new-style

Please note that while you can revert to legacy style
configuration at any time unless you have explicitly
entered new-style configuration, the following caveats
should be carefully read and understood.

(1) If you save the config in this mode, it will be written
    to NVRAM in NEW-style config, and if you subsequently
    reload the router without reverting to legacy config and
    saving that, you will no longer be able to revert.

(2) In this and legacy mode, Webauth is not IPv6-capable. It
    will only become IPv6-capable once you have entered new-
    style config manually, or have reloaded with config saved
    in 'authentication display new' mode.

Device# configure terminal
Device(config)# aaa accounting identity default start-stop group radius
Device(config)# exit
```

# aaa authentication dot1x

To specify one or more authentication, authorization, and accounting (AAA) methods for use on interfaces running IEEE 802.1x, use the **aaa authentication dot1x** command in global configuration mode. To disable authentication, use the **no** form of this command

**aaa authentication dot1x** { **default** *listname* } *method1* [ *method2* . . . ]
**no aaa authentication dot1x** { **default** *listname* } *method1* [ *method2* . . . ]

## Syntax Description

| | |
|---|---|
| **default** | Uses the listed authentication methods that follow this argument as the default list of methods when a user logs in. |
| *listname* | Character string used to name the list of authentication methods tried when a user logs in. |
| *method1* [*method2...*] | A method can be least one of these keywords:<br>• **enable**: Uses the enable password for authentication.<br>• **group radius**: Uses the list of all the RADIUS servers for authentication.<br>• **line**: Uses the line password for authentication.<br>• **local**: Uses the local username database for authentication.<br>• **local-case**: Uses the case-sensitive local username database for authentication.<br>• **none**: Uses no authentication. The client is automatically authenticated by the device without using the information supplied by the client.<br>• **group** *radius-server-group-name*: Uses the group RADIUS server for authentication.<br>• **cache** *radius-server-group-name*: Uses the cache RADIUS server for authentication.<br>**Note**<br>You must configure the AAA authentication method list with both **group** *radius-server-group-name* and **cache** *radius-server-group-name* to use AAA cache-based authentication. For more information, see "Updating Authorization and Authentication Method Lists to Specify How Cache Information is Used" procedure of the "Configuring AAA Authorization and Authentication Cache" configuration guide. |

## Command Default

No authentication is performed.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Cupertino 17.7.1 | This command was modified. The **cache** keyword was introduced. |

## Usage Guidelines

The *method* argument identifies the list of methods that the authentication algorithm runs in the given sequence to validate the password provided by the client. The only method that is truly 802.1x-compliant is the **group radius** method, in which the client data is validated against a RADIUS authentication server. The remaining methods enable AAA to authenticate the client by using locally configured data. For example, the **local** and **local-case** methods use the username and password that are saved in the Cisco IOS configuration file. The **enable** and **line** methods use the **enable** and **line** passwords for authentication.

If you specify **group radius**, you must configure the RADIUS server by entering the **radius server** *server-name* global configuration command. If you are not using a RADIUS server, you can use the **local** or **local-case** methods, which access the local username database to perform authentication. By specifying the **enable** or **line** methods, you can supply the client with a password to provide access to the device.

Use the **show running-config** privileged EXEC command to display the configured lists of authentication methods.

## Examples

The following example shows how to enable AAA and how to create an authentication list for 802.1x:

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa group server radius RASERV
Device(config)# server name RASERV-1
Device(config)# aaa authentication dot1x default group RASERV
```

## Related Commands

| Command | Description |
|---|---|
| **debug dot1x** | Displays 802.1x debugging information. |
| **identity profile default** | Creates an identity profile and enters dot1x profile configuration mode. |
| **show dot1x** | Displays details for an identity profile. |

# aaa authorization

To set the parameters that restrict user access to a network, use the **aaa authorization** command in global configuration mode. To remove the parameters, use the **no** form of this command.

**aaa authorization** { **auth-proxy** | **cache** | **commands** *level* | **config-commands** | **configuration** | **console** | **credential-download** | **exec** | **multicast** | **network** | **reverse-access** | **template** } { **default** | *list_name* } [ *method1* [ *method2* . . . ] ]
**no aaa authorization** { **auth-proxy** | **cache** | **commands** *level* | **config-commands** | **configuration** | **console** | **credential-download** | **exec** | **multicast** | **network** | **reverse-access** | **template** } { **default** | *list_name* } [ *method1* [ *method2* . . . ] ]

## Syntax Description

| | |
|---|---|
| **auth-proxy** | Runs authorization for authentication proxy services. |
| **cache** | Configures the authentication, authorization, and accounting (AAA) server. |
| **commands** | Runs authorization for all commands at the specified privilege level. |
| *level* | Specific command level that should be authorized. Valid entries are 0 through 15. |
| **config-commands** | Runs authorization to determine whether commands entered in configuration mode are authorized. |
| **configuration** | Downloads the configuration from the AAA server. |
| **console** | Enables the console authorization for the AAA server. |
| **credential-download** | Downloads EAP credential from Local/RADIUS/LDAP. |
| **exec** | Enables the console authorization for the AAA server. |
| **multicast** | Downloads the multicast configuration from the AAA server. |
| **network** | Runs authorization for all network-related service requests, including Serial Line Internet Protocol (SLIP), PPP, PPP Network Control Programs (NCPs), and AppleTalk Remote Access (ARA). |
| **reverse-access** | Runs authorization for reverse access connections, such as reverse Telnet. |
| **template** | Enables template authorization for the AAA server. |
| **default** | Uses the listed authorization methods that follow this keyword as the default list of methods for authorization. |
| *list_name* | Character string used to name the list of authorization methods. |
| *method1* [*method2...*] | (Optional) An authorization method or multiple authorization methods to be used for authorization. A method may be any one of the keywords listed in the table below. |

**Command Default**

Authorization is disabled for all actions (equivalent to the method keyword **none**).

**Command Modes**

Global configuration (config)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

**Usage Guidelines**

Use the **aaa authorization** command to enable authorization and to create named methods lists, which define authorization methods that can be used when a user accesses the specified function. Method lists for authorization define the ways in which authorization will be performed and the sequence in which these methods will be performed. A method list is a named list that describes the authorization methods (such as RADIUS or TACACS+) that must be used in sequence. Method lists enable you to designate one or more security protocols to be used for authorization, which ensures a backup system in case the initial method fails. Cisco IOS XE software uses the first method listed to authorize users for specific network services; if that method fails to respond, the Cisco IOS XE software selects the next method listed in the method list. This process continues until there is successful communication with a listed authorization method, or until all the defined methods are exhausted.

**Note** The Cisco IOS XE software attempts authorization with the next listed method only when there is no response from the previous method. If authorization fails at any point in this cycle--meaning that the security server or the local username database responds by denying the user services--the authorization process stops and no other authorization methods are attempted.

If the **aaa authorization** command for a particular authorization type is issued without a specified named method list, the default method list is automatically applied to all interfaces or lines (where this authorization type applies) except those that have a named method list explicitly defined. (A defined method list overrides the default method list.) If no default method list is defined, then no authorization takes place. The default authorization method list must be used to perform outbound authorization, such as authorizing the download of IP pools from the RADIUS server.

Use the **aaa authorization** command to create a list by entering the values for the *list-name* and the *method* arguments, where *list-name* is any character string used to name this list (excluding all method names) and *method* identifies the list of authorization methods tried in the given sequence.

**Note** In the table that follows, the **group***group-name*, **group ldap**, **group radius**, and **group tacacs+** methods refer to a set of previously defined RADIUS or TACACS+ servers. Use the **radius server** and **tacacs server** commands to configure the host servers. Use the **aaa group server radius**, **aaa group server ldap**, and **aaa group server tacacs+** commands to create a named group of servers.

This table describes the method keywords.

***Table 2: aaa authorization Methods***

| Keyword | Description |
|---|---|
| **cache** *group-name* | Uses a cache server group for authorization. |

| Keyword | Description |
|---|---|
| **group** *group-name* | Uses a subset of RADIUS or TACACS+ servers for accounting as defined by the **server group** *group-name* command. |
| **group ldap** | Uses the list of all Lightweight Directory Access Protocol (LDAP) servers for authentication. |
| **group radius** | Uses the list of all RADIUS servers for authentication as defined by the **aaa group server radius** command. |
| **grouptacacs+** | Uses the list of all TACACS+ servers for authentication as defined by the **aaa group server tacacs+** command. |
| **if-authenticated** | Allows the user to access the requested function if the user is authenticated.<br>**Note**<br>The **if-authenticated** method is a terminating method. Therefore, if it is listed as a method, any methods listed after it will never be evaluated. |
| **local** | Uses the local database for authorization. |
| **none** | Indicates that no authorization is performed. |

Cisco IOS XE software supports the following methods for authorization:

- Cache Server Groups—The device consults its cache server groups to authorize specific rights for users.
- If-Authenticated—The user is allowed to access the requested function provided the user has been authenticated successfully.
- Local—The device consults its local database, as defined by the **username** command, to authorize specific rights for users. Only a limited set of functions can be controlled through the local database.
- None—The network access server does not request authorization information; authorization is not performed over this line or interface.
- RADIUS—The network access server requests authorization information from the RADIUS security server group. RADIUS authorization defines specific rights for users by associating attributes, which are stored in a database on the RADIUS server, with the appropriate user.
- TACACS+—The network access server exchanges authorization information with the TACACS+ security daemon. TACACS+ authorization defines specific rights for users by associating attribute-value (AV) pairs, which are stored in a database on the TACACS+ security server, with the appropriate user.

Method lists are specific to the type of authorization being requested. AAA supports five different types of authorization:

- Commands—Applies to the EXEC mode commands a user issues. Command authorization attempts authorization for all EXEC mode commands, including global configuration commands, associated with a specific privilege level.

- EXEC—Applies to the attributes associated with a user EXEC terminal session.
- Network—Applies to network connections. The network connections can include a PPP, SLIP, or ARA connection.
- Reverse Access—Applies to reverse Telnet sessions.
- Configuration—Applies to the configuration downloaded from the AAA server.

When you create a named method list, you are defining a particular list of authorization methods for the indicated authorization type.

Once defined, the method lists must be applied to specific lines or interfaces before any of the defined methods are performed.

The authorization command causes a request packet containing a series of AV pairs to be sent to the RADIUS or TACACS daemon as part of the authorization process. The daemon can do one of the following:

- Accept the request as is.
- Make changes to the request.
- Refuse the request and authorization.

For a list of supported RADIUS attributes, see the module RADIUS Attributes. For a list of supported TACACS+ AV pairs, see the module TACACS+ Attribute-Value Pairs.

**Note** Five commands are associated with privilege level 0: **disable**, **enable**, **exit**, **help**, and **logout**. If you configure AAA authorization for a privilege level greater than 0, these five commands will not be included in the privilege level command set.

The following example shows how to define the network authorization method list named mygroup, which specifies that RADIUS authorization will be used on serial lines using PPP. If the RADIUS server fails to respond, local network authorization will be performed.

```
Device> enable
Device# configure terminal
Device(config)# aaa authorization network mygroup group radius local
Device(config)# exit
```

# aaa common-criteria policy

To configure the AAA common criteria security policies, use the **aaa common-criteria policy** command in global configuration mode. To disable the AAA common criteria policies, use the **no** form of this command.

**aaa common-criteria policy** *policy-name*
**no aaa common-criteria policy** *policy-name*

## Syntax Description

| | |
|---|---|
| *policy-name* | Name of the AAA common criteria security policy. |

## Command Default

The common criteria security policy is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Dublin 17.10.1 | This command was modified. The **character-repetition** and **restrict-consecutive-letters** keywords were introduced. |

## Usage Guidelines

Use the **aaa common-criteria policy** command to enter the common criteria configuration policy mode. To check the available options in this mode, type **?** after entering into common criteria configuration policy mode (config-cc-policy).

The following options are available:

- **char-change**: Change the number of characters between the old and new passwords. The range is from 1 to 64, and the default value is 4.
- **copy**: Copy the common criteria policy parameters from an existing policy.
- **exit**: Exit from common criteria configuration mode.
- **lifetime**: Configure the maximum lifetime of a password by providing the configurable value, in years, months, days, hours, minutes, and seconds. If the lifetime parameter is not configured, the password will never expire.

**Note** The **lifetime** option of the AAA common criteria policy is not supported for the **enable password** command.

- **lower-case**: Number of lowercase characters. The range is from 0 to 64.
- **upper-case**: Number of uppercase characters. The range is from 0 to 64.
- **min-length**: Minimum length of the password. The range is from 1 to 64, and the default value is 1.
- **max-length**: Maximum length of the password. The range is from 1 to 127, and the default value is 127.

- **numeric-count**: Number of numeric characters. The range is from 0 to 64.
- **special-case**: Number of special characters. The range is from 0 to 64.
- **character-repetition**: Maximum number of times a character can repeat consecutively in password. The range is from 2 to 5.
- **restrict-consecutive-letters**: Prohibit consecutive 4 characters or numbers from the keyboard sequentially in either directions.

**Note** When you use the **aaa password restriction** command, the security checks require your password to have atleast one of the four classes. The classes are categorised by uppercase, lowercase, numeric and special character. When you use both **aaa password restriction** and **aaa common-criteria policy** commands together, all the checks are run for the **aaa password restriction** command first and then the common criteria validation takes place.

The character repetition functionality configured under **aaa common-criteria policy** command takes precedence over the **aaa password restriction** command when both are configured together. The character repetition option allows you to choose the count value when you configure under the **aaa common-criteria policy** command.

The **login password-reuse-interval** command cannot store old passwords across device reboots. Using common criteria policy command helps to store five recently changed passwords across device reboots.

## Examples

The following example shows how to create a common criteria security policy:

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa common-criteria policy policy1
Device(config-cc-policy)# end
```

## Related Commands

| Command | Description |
|---|---|
| **aaa new-model** | Enables AAA access control model. |
| **debug aaa common-criteria** | Enables debugging for AAA common criteria password security policies. |
| **show aaa common-criteria policy** | Displays common criteria security policy details. |

# aaa new-model

To enable the authentication, authorization, and accounting (AAA) access control model, issue the **aaa new-model** command in global configuration mode. To disable the AAA access control model, use the **no** form of this command.

**aaa new-model**
**no aaa new-model**

## Syntax Description

This command has no arguments or keywords.

## Command Default

AAA is not enabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

This command enables the AAA access control system.

If the **login local** command is configured for a virtual terminal line (VTY), and the **aaa new-model** command is removed, you must reload the switch to get the default configuration or the **login** command. If the switch is not reloaded, the switch defaults to the **login local** command under the VTY.

**Note** We do not recommend removing the **aaa new-model** command. This command is required for dot1x.

## Examples

The following example initializes AAA:

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# exit
```

The following example shows a VTY configured and the **aaa new-model** command removed:

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# line vty 0 15
Device(config-line)# login local
Device(config-line)# exit
Device(config)# no aaa new-model
Device(config)# exit
Device# show running-config | b line vty

line vty 0 4
 login local  !<=== Login local instead of "login"
line vty 5 15
 login local
```

!

## Related Commands

| Command | Description |
|---|---|
| **aaa accounting** | Enables AAA accounting of requested services for billing or security purposes. |
| **aaa authentication arap** | Enables an AAA authentication method for ARAP using TACACS+. |
| **aaa authentication enable default** | Enables AAA authentication to determine if a user can access the privileged command level. |
| **aaa authentication login** | Sets AAA authentication at login. |
| **aaa authentication ppp** | Specifies one or more AAA authentication method for use on serial interfaces running PPP. |
| **aaa authorization** | Sets parameters that restrict user access to a network. |

# access-session mac-move deny

To disable MAC move on a device, use the **access-session mac-move deny** global configuration command. To return to the default setting, use the **no** form of this command.

**access-session mac-move deny**
**no access-session mac-move deny**

## Syntax Description

This command has no arguments or keywords.

## Command Default

MAC move is enabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **no** form of this command enables authenticated hosts to move between any authentication-enabled ports (MAC authentication bypass [MAB], 802.1x, or Web-auth) on a device. For example, if there is a device between an authenticated host and port, and that host moves to another port, the authentication session is deleted from the first port, and the host is reauthenticated on the new port.

If MAC move is disabled, and an authenticated host moves to another port, it is not reauthenticated, and a violation error occurs.

This example shows how to enable MAC move on a device:

```
Device> enable
Device# configure terminal
Device(config)# no access-session mac-move deny
Device(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **authentication event** | Sets the action for specific authentication events. |
| **authentication fallback** | Configures a port to use web authentication as a fallback meth authentication. |
| **authentication host-mode** | Sets the authorization manager mode on a port. |
| **authentication open** | Enables or disables open access on a port. |
| **authentication order** | Sets the order of authentication methods used on a port. |
| **authentication periodic** | Enables or disables reauthentication on a port. |
| **authentication port-control** | Enables manual control of the port authorization state. |
| **authentication priority** | Adds an authentication method to the port-priority list. |

| Command | Description |
| --- | --- |
| **authentication timer** | Configures the timeout and reauthentication parameters fo |
| **authentication violation** | Configures the violation modes that occur when a new dev to a port with the maximum number of devices already co |
| **show authentication** | Displays information about authentication manager events |

# access-session host-mode multi-host

To allow hosts to gain access to a controlled port only after the first client is authenticated, use the **access-session host-mode multi-host** command in interface configuration mode. To return to the default value, use the **no** form of this command.

**access-session host-mode multi-host** [ **peer** ]
**no access-session host-mode multi-host** [ **peer** ]

## Syntax Description

| | |
|---|---|
| **peer** | Specifies that only a peer device can be authenticated first. |

## Command Default

Access to a port is multi-auth, wherein multiple clients can be authenticated on the port.

## Command Modes

Interface Configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Cupertino 17.7.1 | The keyword **peer** was added. |

## Usage Guidelines

Before you use this command, you must enable the **access-session port-control auto** command.

In multi-host mode, only one of the attached hosts has to be successfully authorized for all hosts to be granted network access. If the port becomes unauthorized (reauthentication fails or an Extensible Authentication Protocol over LAN (EAPOL) logoff message is received), all attached clients are denied access to the network.

Starting Cisco IOS XE Release 17.7.1, you can enable a peer device to be authenticated first, using the **access-session host-mode multi-host peer** command. While this command is especially useful in SD-Access networks, it is also applicable to non-SD-Access environments.

Consider a Cisco SD-Access fabric network where an extended node and its clients have to be securely onboarded. We must ensure that until the extended node is authenticated, the clients connected to it do not have access to the network. In such a case, use the **access-session host-mode multi-host peer** command to authenticate the extended node first. (The extended node is the peer device that is connected to the authenticator port.) Cisco ISE pushes this CLI through an interface template that is applied to the fabric edge node for IEEE 802.1X authentication. A change in the host mode clears all the existing sessions on the fabric edge. We recommend enabling the **access-session interface-template sticky timer** command in the global configuration mode to avoid the template from getting unbound from the edge node port. The sticky timer value should be a minimum of 60 seconds to avoid the bind–unbind loop issues. The interface template is unbound after the sticky timer expires.

Similarly, in cases where trunk ports are connected to the access device, use the **access-session host-mode multi-host peer** command to authenticate only the peer MAC. This avoids authenticating all the MAC addresses learnt.

**Note**

- The keyword **peer** is supported only in the fabric edge mode. It is not supported in the legacy mode.
- The peer configuration clears all the existing sessions on the authenticator port.
- A session will not be established by the device unless it receives CDP, LLDP, or EAPOL packets from the peer.

You can use the **show access-session interface** command to verify the port setting.

### Example

The following example shows how to enable authorization of only the peer device on port1/0/2.

```
Device# configure terminal
 Device(config)# interface GigabitEthernet 1/0/2
 Device(config-if)# access-session host-mode multi-host peer
 Device(config-if)# access-session closed
 Device(config-if)# access-session port-control auto
```

## Related Commands

| | |
|---|---|
| **access-session closed** | Prevents preauthentication access on a port. |
| **access-session port-control** | Sets the authorization state of a port. |
| **show access-session** | Displays information about authentication sessions. |

# action

To set the action for the VLAN access map entry, use the **action** command in access-map configuration mode. To return to the default setting, use the **no** form of this command.

**action** {**drop** | **forward**}
**no action**

## Syntax Description

| | |
|---|---|
| **drop** | Drops the packet when the specified conditions are matched. |
| **forward** | Forwards the packet when the specified conditions are matched. |

## Command Default

The default action is to forward packets.

## Command Modes

Access-map configuration (config-access-map)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

You enter access-map configuration mode by using the **vlan access-map** global configuration command.

If the action is **drop**, you should define the access map, including configuring any access control list (ACL) names in match clauses, before applying the map to a VLAN, or all packets could be dropped.

In access-map configuration mode, use the **match access-map** configuration command to define the match conditions for a VLAN map. Use the **action** command to set the action that occurs when a packet matches the conditions.

The drop and forward parameters are not used in the **no** form of the command.

You can verify your settings by entering the **show vlan access-map** privileged EXEC command.

## Examples

This example shows how to identify and apply a VLAN access map (vmap4) to VLANs 5 and 6 that causes the VLAN to forward an IP packet if the packet matches the conditions defined in access list al2:

```
Device> enable
Device# configure terminal
Device(config)# vlan access-map vmap4
Device(config-access-map)# match ip address al2
Device(config-access-map)# action forward
Device(config-access-map)# exit
Device(config)# vlan filter vmap4 vlan-list 5-6
Device(config)# exit
```

# api-key (Parameter Map)

To configure the application programming interface (API) key used for authorization during device registration, use the **api-key** command in parameter-map type inspect configuration mode. To remove the key, use the **no** form of this command.

**api-key** *value*
**no api-key**

## Syntax Description

| | |
|---|---|
| *value* | The API key. You can obtain this from the Cisco Umbrella registration server. |

## Command Default

No key is created for the parameter map.

## Command Modes

Parameter-map type inspect configuration (config-profile)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Cupertino 17.7.1 | This command was introduced. |

## Usage Guidelines

To perform API registration for the Umbrella Switch Connector, the **api-key**, **orgid** and **secret** commands must be configured one after the other. The values for these commands can be retrieved from the Cisco Umbrella registration server.

## Examples

The following example shows how to perform API registration for the Umbrella Switch Connector:

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type umbrella global
Device(config-profile)# api-key 5f22922exxxxxxxxx51174af822734
Device(config-profile)# orgid 26xxx16
Device(config-profile)# secret 0 a0d176ebxxxxxxxfbb343dfc4fd209
Device(config-profile)# end
```

## Related Commands

| Command | Description |
|---|---|
| **parameter-map type umbrella global** | Configures a parameter-map type in umbrella mode. |

# authentication host-mode

To set the authorization manager mode on a port, use the **authentication host-mode** command in interface configuration mode. To return to the default setting, use the **no** form of this command.

**authentication host-mode** {**multi-auth** | **multi-domain** | **multi-host** | **single-host**}
**no authentication host-mode**

## Syntax Description

| | |
|---|---|
| **multi-auth** | Enables multiple-authorization mode (multi-auth mode) on the port. |
| **multi-domain** | Enables multiple-domain mode on the port. |
| **multi-host** | Enables multiple-host mode on the port. |
| **single-host** | Enables single-host mode on the port. |

## Command Default

Single host mode is enabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Single-host mode should be configured if only one data host is connected. Do not connect a voice device to authenticate on a single-host port. Voice device authorization fails if no voice VLAN is configured on the port.

Multi-domain mode should be configured if data host is connected through an IP phone to the port. Multi-domain mode should be configured if the voice device needs to be authenticated.

Multi-auth mode should be configured to allow devices behind a hub to obtain secured port access through individual authentication. Only one voice device can be authenticated in this mode if a voice VLAN is configured.

Multi-host mode also offers port access for multiple hosts behind a hub, but multi-host mode gives unrestricted port access to the devices after the first user gets authenticated.

This example shows how to enable multi-auth mode on a port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# authentication host-mode multi-auth
Device(config-if)# end
```

This example shows how to enable multi-domain mode on a port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# authentication host-mode multi-domain
Device(config-if)# end
```

This example shows how to enable multi-host mode on a port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# authentication host-mode multi-host
Device(config-if)# end
```

This example shows how to enable single-host mode on a port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# authentication host-mode single-host
Device(config-if)# end
```

You can verify your settings by entering the **show authentication sessions interface** *interface* **details** privileged EXEC command.

# authentication logging verbose

To filter detailed information from authentication system messages, use the **authentication logging verbose** command in global configuration mode on the switch stack or on a standalone switch.

**authentication logging verbose**
**no authentication logging verbose**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Detailed logging of system messages is not enabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

This command filters details, such as anticipated success, from authentication system messages. Failure messages are not filtered.

To filter verbose authentication system messages:

```
Device> enable
Device# configure terminal
Device(config)# authentication logging verbose
Device(config)# exit
```

You can verify your settings by entering the **show running-config** privileged EXEC command.

## Related Commands

| Command | Description |
|---|---|
| **authentication logging verbose** | Filters details fro |
| **dot1x logging verbose** | Filters details fro |
| **mab logging verbose** | Filters details fro |

# authentication mac-move permit

To enable MAC move on a device, use the **authentication mac-move permit** command in global configuration mode. To disable MAC move, use the **no** form of this command.

**authentication mac-move permit**
**no authentication mac-move permit**

## Syntax Description

This command has no arguments or keywords.

## Command Default

MAC move is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The command enables authenticated hosts to move between any authentication-enabled ports (MAC authentication bypass [MAB], 802.1x, or Web-auth) on a device. For example, if there is a device between an authenticated host and port, and that host moves to another port, the authentication session is deleted from the first port, and the host is reauthenticated on the new port.

If MAC move is disabled, and an authenticated host moves to another port, it is not reauthenticated, and a violation error occurs.

This example shows how to enable MAC move on a device:

```
Device> enable
Device# configure terminal
Device(config)# authentication mac-move permit
Device(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **access-session mac-move deny** | Disables MAC move on a device. |
| **authentication event** | Sets the action for specific authentication events |
| **authentication fallback** | Configures a port to use web authentication as a IEEE 802.1x authentication. |
| **authentication host-mode** | Sets the authorization manager mode on a port. |
| **authentication open** | Enables or disables open access on a port. |
| **authentication order** | Sets the order of authentication methods used or |
| **authentication periodic** | Enable or disables reauthentication on a port. |
| **authentication port-control** | Enables manual control of the port authorization |

| Command | Description |
|---|---|
| **authentication priority** | Adds an authentication method to the port-priority l |
| **authentication timer** | Configures the timeout and reauthentication parame |
| **authentication violation** | Configures the violation modes that occur when a n device connects to a port with the maximum numbe |
| **show authentication** | Displays information about authentication manager |

# authentication priority

To add an authentication method to the port-priority list, use the **authentication priority** command in interface configuration mode. To return to the default, use the **no** form of this command.

**authentication priority** [**dot1x** | **mab**] {**webauth**}
**no authentication priority** [**dot1x** | **mab**] {**webauth**}

## Syntax Description

| | |
|---|---|
| **dot1x** | (Optional) Adds 802.1x to the order of authentication methods. |
| **mab** | (Optional) Adds MAC authentication bypass (MAB) to the order of authenti methods. |
| **webauth** | Adds web authentication to the order of authentication methods. |

## Command Default

The default priority is 802.1x authentication, followed by MAC authentication bypass and web authentication.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Ordering sets the order of methods that the device attempts when trying to authenticate a new device is connected to a port.

When configuring multiple fallback methods on a port, set web authentication (webauth) last.

Assigning priorities to different authentication methods allows a higher-priority method to interrupt an in-progress authentication method with a lower priority.

**Note** If a client is already authenticated, it might be reauthenticated if an interruption from a higher-priority method occurs.

The default priority of an authentication method is equivalent to its position in execution-list order: 802.1x authentication, MAC authentication bypass (MAB), and web authentication. Use the **dot1x**, **mab**, and **webauth** keywords to change this default order.

This example shows how to set 802.1x as the first authentication method and web authentication as the second authentication method:

```
Device(config-if)# authentication priority dot1x webauth
```

This example shows how to set MAB as the first authentication method and web authentication as the second authentication method:

```
Device> enable
Device# configure terminal
```

```
Device(config)# interface gigabitethernet 0/1/2
Device(config-if)# authentication priority mab webauth
Device(config-if)# end
```

## Related Commands

| Command | Description |
|---|---|
| **authentication control-direction** | Configures the port mode as unidirectional or bidirectional. |
| **authentication event fail** | Specifies how the Auth Manager handles authentication failures as a r |
| **authentication event no-response action** | Specifies how the Auth Manager handles authentication failures as a r |
| **authentication event server alive action reinitialize** | Reinitializes an authorized Auth Manager session when a previously u and accounting server becomes available. |
| **authentication event server dead action authorize** | Authorizes Auth Manager sessions when the authentication, authoriza unreachable. |
| **authentication fallback** | Enables a web authentication fallback method. |
| **authentication host-mode** | Allows hosts to gain access to a controlled port. |
| **authentication open** | Enables open access on a port. |
| **authentication order** | Specifies the order in which the Auth Manager attempts to authentica |
| **authentication periodic** | Enables automatic reauthentication on a port. |
| **authentication port-control** | Configures the authorization state of a controlled port. |
| **authentication timer inactivity** | Configures the time after which an inactive Auth Manager session is t |
| **authentication timer reauthenticate** | Specifies the period of time between which the Auth Manager attemp |
| **authentication timer restart** | Specifies the period of time after which the Auth Manager attempts to |
| **authentication violation** | Specifies the action to be taken when a security violation occurs on a |
| **mab** | Enables MAC authentication bypass on a port. |
| **show authentication registrations** | Displays information about the authentication methods that are registe |
| **show authentication sessions** | Displays information about current Auth Manager sessions. |
| **show authentication sessions interface** | Displays information about the Auth Manager for a given interface. |

# authentication timer reauthenticate

To specify the period of time between which the Auth Manager attempts to reauthenticate authorized ports, use the **authenticationtimerreauthenticate** command in interface configuration or template configuration mode. To reset the reauthentication interval to the default, use the **no** form of this command.

**authentication timer reauthenticate** { *seconds* | **server** }

**no authentication timer reauthenticate**

## Syntax Description

| | |
|---|---|
| *seconds* | The number of seconds between reauthentication attempts. The range is from 1 to 1073741823. The default is 3600 seconds. |
| **server** | Specifies that the interval between reauthentication attempts is defined by the Session-Timeout value (RADIUS Attribute 27) on the authentication, authorization, and accounting (AAA) server. |

## Command Default

The automatic reauthentication interval is set to 3600 seconds.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced |
| Cisco IOS XE Bengaluru 17.5.1 | The supported time-out range was increased from 65535 seconds to 1073741823 seconds |

## Usage Guidelines

Use the command **authenticationtimer reauthenticate** command to set the automatic reauthentication interval of an authorized port. If you use the **authenticationtimerinactivity** command to configure an inactivity interval, configure the reauthentication interval to be longer than the inactivity interval.

In releases prior to Cisco IOS XE Bengaluru 17.5.1, the supported timeout range is 1 to 65535 seconds. While downgrading from or releases after Cisco IOS XE Bengaluru 17.5.1 set the configuration timeout to supported values to avoid ISSD breakage.

## Examples

The following example shows how to set the reauthentication interval on a port to 1800 seconds:

```
Device >enable
Device #configure terminal
Device(config)#interface gigabitethernet2/0/1
Device(config-if)#authentication timer reauthenticate 1800
Device(config-if)#end
```

## Related Commands

| Command | Description |
|---|---|
| **authenticationperiodic** | Enables automatic reauthentication. |
| **authenticationtimerinactivity** | Specifies the interval after which the Auth Manager ends an inactive session. |

| Command | Description |
| --- | --- |
| **authenticationtimerrestart** | Specifies the interval after which the Auth Manager attempts to authenticate an unauthorized port. |

# authentication violation

To configure the violation modes that occur when a new device connects to a port or when a new device connects to a port after the maximum number of devices are connected to that port, use the **authentication violation** command in interface configuration mode.

**authentication violation** { **protect** | **replace** | **restrict** | **shutdown** }
**no authentication violation** { **protect** | **replace** | **restrict** | **shutdown** }

## Syntax Description

| | |
|---|---|
| **protect** | Drops unexpected incoming MAC addresses. No syslog errors are generated. |
| **replace** | Removes the current session and initiates authentication with the new host. |
| **restrict** | Generates a syslog error when a violation error occurs. |
| **shutdown** | Error-disables the port or the virtual port on which an unexpected MAC address occurs. |

## Command Default

Authentication violation shutdown mode is enabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **authentication violation** command to specify the action to be taken when a security violation occurs on a port.

This example shows how to configure an IEEE 802.1x-enabled port as error-disabled and to shut down when a new device connects it:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# authentication violation shutdown
Device(config-if)# end
```

This example shows how to configure an 802.1x-enabled port to generate a system error message and to change the port to restricted mode when a new device connects to it:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# authentication violation restrict
Device(config-if)# end
```

This example shows how to configure an 802.1x-enabled port to ignore a new device when it connects to the port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# authentication violation protect
Device(config-if)# end
```

This example shows how to configure an 802.1x-enabled port to remove the current session and initiate authentication with a new device when it connects to the port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/1
Device(config-if)# authentication violation replace
Device(config-if)# end
```

You can verify your settings by entering the **show running-config interface** *interface-name* command.

# cisp enable

To enable Client Information Signaling Protocol (CISP) on a device so that it acts as an authenticator to a supplicant device and a supplicant to an authenticator device, use the **cisp enable** global configuration command.

**cisp enable**
**no cisp enable**

## Syntax Description

This command has no arguments or keywords.

## Command Default

No default behavior or values.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The link between the authenticator and supplicant device is a trunk. When you enable VTP on both devices, the VTP domain name must be the same, and the VTP mode must be server.

To avoid the MD5 checksum mismatch error when you configure VTP mode, verify that:

- VLANs are not configured on two different devices, which can be caused by two VTP servers in the same domain.
- Both devices have different configuration revision numbers.

This example shows how to enable CISP:

```
Device> enable
Device# configure terminal
Device(config)# cisp enable
Device(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x credentials***profile* | Configures a profile on a supplicant device. |
| **dot1x supplicant force-multicast** | Forces 802.1X supplicant to send multicast pac |
| **dot1x supplicant controlled transient** | Configures controlled access by 802.1X suppli |
| **show cisp** | Displays CISP information for a specified inter |

# clear aaa cache group

To clear an individual entry or all entries in the cache, use the **clear aaa cache group** command in privileged EXEC mode.

**clear aaa cache group** *name* { **profile** *name* | **all** }

## Syntax Description

| | |
|---|---|
| *name* | Text string representing the name of a cache server group. |
| **profile** *name* | Specifies the name of an individual profile entry that must be cleared. |
| **all** | Specifies that all the profiles in the named cache group be cleared. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

To update an old record with profile cache settings and to remove an old record from the cache, clear the cache for the profile.

## Examples

The following example shows how to clear all the cache entries in the localusers group:

```
Device# clear aaa cache group localusers all
```

## Related Commands

| Command | Description |
|---|---|
| **show aaa cache group** | Displays all the cache entries stored by the AAA cache. |

# clear device-tracking database

To delete device-tracking database (binding table) entries, and clear counters, events, and messages, enter the **clear device-tracking** command in privileged EXEC mode.

**clear device-tracking** { **counters** [ **interface** *inteface_type_no* | **vlan** *vlan_id* ] | **database** [ **address** { *hostname* | **all** } [ **interface** *inteface_type_no* | **policy** *policy_name* | **vlan** *vlan_id* ] | **interface** *inteface_type_no* [ **vlan** *vlan_id* ] | **mac** *mac_address* [ **interface** *inteface_type_no* | **policy** *policy_name* | **vlan** *vlan_id* ] | **policy** *policy_name* | **prefix** { *prefix* | **all** } [ **interface** *inteface_type_no* | **policy** *policy_name* | **vlan** *vlan_id* ] | **vlanid** *vlan_id* ] | **events** | **messages** }

## Syntax Description

| | |
|---|---|
| **counters** | Clears device-tracking counters for the specified interface or VLAN.<br>Counters are displayed in the **show device-tracking counters all** privileged EXEC command. |
| **interface** *inteface_type_no* | Enter an interface type and number. Use the question mark (?) online help function to display the types of interfaces available on the device.<br>The clear action is performed for the interface you specify. |
| **vlan** *vlan_id* | Enter a VLAN ID. The clear action is performed for the VLAN ID you specify.<br>The valid value range is from 1 to 4095. |
| **database** | Clears dynamic entries in the binding table.<br>**Note**<br>Static entries configured by using the **device-tracking binding vlan** *vlan_id* command are not deleted.<br>You can delete all the dynamic entries in the table, or optionally, you can specify one or more IP addresses, MAC addresses, IPv6 prefixes, entries on a particular interface or VLAN, or a policy. |
| *hostname* | Enter the hostname or IP address on which you want to perform the clear action. |
| **all** | Performs the clear action on all IP addresses or IPv6 prefixes. |
| **policy** *policy_name* | Performs the clear action on the specified policy. Enter the policy name. |
| **mac** *mac_address* | Performs the clear action on the specified MAC address. Enter the MAC address. |
| **prefix** *prefix* | Performs the clear action on the specified IPv6 prefix. Enter a prefix or enter **all** to indicate all prefixes. |
| **events** | Clears the device-tracking events history.<br>Events are displayed in the **show device-tracking events** privileged EXEC command. |
| **messages** | Clears the device-tracking message history.<br>Events are displayed in the **show device-tracking messages** privileged EXEC command. |

**Command Default**

Database entries go through their binding entry lifecycle.

Counters: Each counter is a nonnegative 32-bit integer and it wraps-around when the limit is reached.

Events and messages: After the limit of 255 is reached, starting with the oldest, events and messages are overwritten.

**Command Modes**

Privileged EXEC (#)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

### Examples

The following example shows you how to clear all entries from the binding table.

```
Device# show device-tracking database Binding Table has 25 entries, 25 dynamic (limit 200000)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


    Network Layer Address                  Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.49                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  699 s
ARP 192.0.9.48                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  691 s
ARP 192.0.9.47                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  687 s
ARP 192.0.9.46                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  714 s
ARP 192.0.9.45                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  692 s
ARP 192.0.9.44                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  702 s
ARP 192.0.9.43                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  680 s
ARP 192.0.9.42                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  708 s
ARP 192.0.9.41                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  683 s
ARP 192.0.9.40                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  708 s
ARP 192.0.9.39                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  710 s
ARP 192.0.9.38                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  697 s
ARP 192.0.9.37                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  707 s
ARP 192.0.9.36                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  695 s
ARP 192.0.9.35                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  708 s
ARP 192.0.9.34                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  706 s
```

```
ARP 192.0.9.33                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  683 s
ARP 192.0.9.32                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  697 s
ARP 192.0.9.31                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  683 s
ARP 192.0.9.30                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  678 s
ARP 192.0.9.29                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  696 s
ARP 192.0.9.28                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  704 s
ARP 192.0.9.27                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  713 s
ARP 192.0.9.26                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  695 s
ARP 192.0.9.25                                 001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  686 s
```

```
Device# clear device-tracking database

*Dec 13 15:10:22.837: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.49 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.838: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.48 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.838: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.47 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.838: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.46 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.839: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.45 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.839: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.44 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.839: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.43 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.839: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.42 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.840: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.41 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.840: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.40 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.840: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.39 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.841: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.38 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.841: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.37 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.841: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.36 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.842: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.35 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.842: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.34 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.842: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.33 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.842: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.32 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.843: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.31 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.843: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.30 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.843: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.29 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.844: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.28 VLAN=200
```

```
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.844: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.27 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.844: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.26 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.844: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.25 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF

Device# show device-tracking database
<no output; binding table cleared>
```

# clear errdisable interface vlan

To reenable a VLAN that was error-disabled, use the **clear errdisable interface** command in privileged EXEC mode.

**clear errdisable interface** *interface-id* **vlan** [*vlan-list*]

## Syntax Description

| | |
|---|---|
| *interface-id* | Specifies an interface. |
| *vlan list* | (Optional) Specifies a list of VLANs to be reenabled. I |

## Command Default

No default behavior or values.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

You can reenable a port by using the **shutdown** and **no shutdown** interface configuration commands, or you can clear error-disable for VLANs by using the **clear errdisable** interface command.

## Examples

This example shows how to reenable all VLANs that were error-disabled on Gigabit Ethernet port 4/0/2:

```
Device# clear errdisable interface gigabitethernet4/0/2 vlan
```

## Related Commands

| Command | Description |
|---|---|
| **errdisable detect cause** | Enables error-disabled detectio |
| **errdisable recovery** | Configures the recovery mecha |
| **show errdisable detect** | Displays error-disabled detecti |
| **show errdisable recovery** | Displays error-disabled recover |
| **show interfaces status err-disabled** | Displays interface status of a li |

# clear fqdn

To clear the fully qualified domain name (FQDN) local cache entries, use the **clear fqdn** command in privileged EXEC mode.

**clear fqdn** { **database** { **all** | **fqdn** *name* [ **ipv4** *address* | **ipv6** *address* ] } | **packet statistics** }

## Syntax Description

| | |
|---|---|
| **database all** | Clears all the FQDN local cache entries. |
| **database fqdn** *name* | Clears the specified FQDN local cache entry. |
| **ipv4** *address* | Clears a particular IP binding matched to the FQDN name. |
| **ipv6** *address* | Clears a particular IPv6 binding matched to the FQDN name. |
| **packet statistics** | Resets all the FQDN packet statistics counters to 0. |

## Command Default

No default behavior or values.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Bengaluru 17.5.1 | This command was introduced. |
| Cisco IOS XE Bengaluru 17.6.1 | This command was modified. The **ipv6** keyword was introduced. |

## Examples

The following example shows how to clear all the FQDN local cache entries:

```
Device> enable
Device# clear fqdn database all
```

The following example shows how to clear a particular IPv4 binding matched to an FQDN name:

```
Device> enable
Device# clear fqdn database fqdn 123.cisco.com ipv4 10.102.103.10
```

The following example shows how to clear a particular IPv6 binding matched to an FQDN name:

```
Device> enable
Device# clear fqdn database fqdn 123.cisco.com ipv6 2001:DB8::1
```

# clear mac address-table

To delete from the MAC address table a specific dynamic address, all dynamic addresses on a particular interface, all dynamic addresses on stack members, or all dynamic addresses on a particular VLAN, use the **clear mac address-table** command in privileged EXEC mode. This command also clears the MAC address notification global counters.

**clear mac address-table** {**dynamic** [**address** *mac-addr* | **interface** *interface-id* | **vlan** *vlan-id*] | **move update** | **notification**}

## Syntax Description

| | |
|---|---|
| **dynamic** | Deletes all dynamic MAC addresses. |
| **address** *mac-addr* | (Optional) Deletes the specified dynamic MAC addr |
| **interface** *interface-id* | (Optional) Deletes all dynamic MAC addresses on t |
| **vlan** *vlan-id* | (Optional) Deletes all dynamic MAC addresses for t |
| **move update** | Clears the MAC address table move-update counters |
| **notification** | Clears the notifications in the history table and reset |

## Command Default

No default behavior or values.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

You can verify that the information was deleted by entering the **show mac address-table** command.

This example shows how to remove a specific MAC address from the dynamic address table:

```
Device> enable
Device# clear mac address-table dynamic address 0008.0070.0007
```

## Related Commands

| Command | Description |
|---|---|
| **mac address-table notification** | Enables the MAC address notification feature. |
| **mac address-table move update** {**receive** \| **transmit**} | Configures MAC address-table move update on the device. |
| **show mac address-table** | Displays the MAC address table static and dynamic entries. |
| **show mac address-table move update** | Displays the MAC address-table move update information on the device. |

| Command | Description |
| --- | --- |
| **show mac address-table notification** | Displays the MAC address notification settings for all interfaces or on the specified interface when the **interface** keyword is appended. |
| **snmp trap mac-notification change** | Enables the SNMP MAC address notification trap on a specific interface. |

# confidentiality-offset

To enable MACsec Key Agreement protocol (MKA) to set the confidentiality offset for MACsec operations, use the **confidentiality-offset** command in MKA-policy configuration mode. To disable confidentiality offset, use the **no** form of this command.

**confidentiality-offset**
**no confidentiality-offset**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Confidentiality offset is disabled.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Examples

The following example shows how to enable the confidentiality offset:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# confidentiality-offset
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **delay-protection** | Configures MKA to use delay protection in sending MKPDU. |
| **include-icv-indicator** | Includes ICV indicator in MKPDU. |
| **key-server** | Configures MKA key-server options. |
| **macsec-cipher-suite** | Configures cipher suite for deriving SAK. |
| **sak-rekey** | Configures the SAK rekey interval. |
| **send-secure-announcements** | Configures MKA to send secure announcements in sending MKPDUs. |
| **ssci-based-on-sci** | Computes SSCI based on the SCI. |
| **use-updated-eth-header** | Uses the updated Ethernet header for ICV calculation. |

# crypto pki trustpool import

To manually import (download) a certificate authority (CA) certificate bundle into a public key infrastructure (PKI) trustpool to update or replace an existing CA bundle, use the **crypto pki trustpool import** command in global configuration mode. To remove any of the configured parameters, use the **no** form of this command.

**crypto pki trustpool import** {**ca-bundle** | **clean** [**terminal** | **url** *url*] | **terminal** | **url** *url*}
**no crypto pki trustpool import** {**ca-bundle** | **clean** [**terminal** | **url** *url*] | **terminal** | **url** *url*}

## Syntax Description

| | |
|---|---|
| **ca-bundle** | Imports a CA certificate bundle configured in the trustpool policy. |
| **clean** | Removes all the downloaded PKI trustpool certificates before the new certificates are downloaded. Use the optional **terminal** keyword to remove the existing CA certificate bundle terminal setting, or the **url** keyword and *url* argument to remove the URL file system setting. |
| **terminal** | Import a CA certificate bundle through the terminal (cut-and-paste) in privacy-enhanced mail (PEM) format. |
| **url** *url* | Imports a CA certificate bundle through the specified URL. |

## Command Default

The PKI trustpool feature is enabled. The device uses the built-in CA certificate bundle in the PKI trustpool, which is updated automatically.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Usage Guidelines

PKI trustpool certificates are automatically updated. When the PKI trustpool certificates are not current, use the **crypto pki trustpool import** command to update them from another location.

**Note** Security threats, as well as the cryptographic technologies that help protect against them, are constantly changing. For more information about the latest Cisco cryptographic recommendations, see the Next Generation Cryptography white paper.

The *url* argument specifies or changes the URL file system of the CA. The following table lists the available URL file systems.

***Table 3: URL File Systems***

| File System | Description |
|---|---|
| **archive:** | Imports from the archive file system. |
| **cns:** | Imports from the Cluster Namespace (CNS) file system. |
| **disk0:** | Imports from the disc0 file system. |

| File System | Description |
|---|---|
| **disk1:** | Imports from the disc1 file system. |
| **ftp:** | Imports from the FTP file system. |
| **http:** | Imports from the HTTP file system. The URL must be in the following formats:<br>• http://*CAname*:80, where *CAname* is the Domain Name System (DNS)<br>• http://*ipv4-address*:80, for example: http://10.10.10.1:80.<br>• http://*[ipv6-address]*:80, for example: http://[2001:DB8:1:1::1]:80. The IPv6 address is a hexadecimal notation, and must be enclosed within brackets in the URL. |
| **https:** | Imports from the HTTPS file system. The URL must use the same format as the HTTP: file system format. |
| **null:** | Imports from the null file system. |
| **nvram:** | Imports from the NVRAM file system. |
| **pram:** | Imports from the Parameter Random-access Memory (PRAM) file system. |
| **rcp:** | Imports from the remote copy protocol (RCP) file system. |
| **scp:** | Imports from the secure copy protocol (SCP) file system. |
| **snmp:** | Imports from the Simple Network Management Protocol (SNMP). |
| **system:** | Imports from the system file. |
| **tar:** | Imports from the UNIX TAR file system. |
| **tftp:** | Imports from the TFTP file system.<br>**Note**<br>The URL must be in the form: tftp://*CAname/filespecification*. |
| **tmpsys:** | Imports from the Cisco IOS tmpsys file system. |
| **unix:** | Imports from the UNIX file system. |
| **xmodem:** | Imports from the xmodem simple file transfer protocol system. |
| **ymodem:** | Imports from the ymodem simple file transfer protocol system. |

## Examples

The following example shows how to remove all the downloaded PKI trustpool CA certificates and subsequently update the CA certificates in the PKI trustpool by downloading a new CA certification bundle:

```
Device> enable
Device# configure terminal
Device(config)# crypto pki trustpool import clean
```

The following example shows how to update all the CA certificates in the PKI trustpool by downloading a new CA certification bundle without removing all the downloaded PKI trustpool CA certificates:

```
Device(config)# crypto pki trustpool import url http://www.cisco.com/security/pki/trs/ios.p7b
```

## Related Commands

| Command | Description |
|---|---|
| **crypto pki trustpool policy** | Configures PKI trustpool policy parameters. |
| **show crypto pki trustpool** | Displays the PKI trustpool certificates of the device, and optionally shows the PKI trustpool policy. |

# debug aaa cache group

To debug the caching mechanism and ensure that caching entries are cached from AAA server responses and found when queried, use the **debug aaa cache group** command in privileged EXEC mode.

**debug aaa cache group**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Debug information for all the cached entries is displayed.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use this command to display debug information about cached entries.

## Examples

The following example displays the debug information about all the cached entries:

```
Device# debug aaa cache group
```

## Related Commands

| Command | Description |
|---|---|
| **clear aaa cache group** | Clears an individual entry or all the entries in the cache. |
| **show aaa cache group** | Displays cache entries stored by the AAA cache. |

# debug aaa dead-criteria transaction

To display authentication, authorization, and accounting (AAA) dead-criteria transaction values, use the **debugaaadead-criteriatransaction**command in privileged EXEC mode. To disable dead-criteria debugging, use the **no** form of this command.

**debug aaa dead-criteria transaction**
**no debug aaa dead-criteria transaction**

## Syntax Description

This command has no arguments or keywords.

## Command Default

If the command is not configured, debugging is not turned on.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Dead-criteria transaction values may change with every AAA transaction. Some of the values that can be displayed are estimated outstanding transaction, retransmit tries, and dead-detect intervals. These values are explained in the table below.

## Examples

The following example shows dead-criteria transaction information for a particular server group:

```
Device> enable
Device# debug aaa dead-criteria transaction

AAA Transaction debugs debugging is on
*Nov 14 23:44:17.403: AAA/SG/TRANSAC: Computed Retransmit Tries: 10, Current Tries: 3,
Current Max Tries: 10
*Nov 14 23:44:17.403: AAA/SG/TRANSAC: Computed Dead Detect Interval: 10s, Elapsed Time:
317s, Current Max Interval: 10s
*Nov 14 23:44:17.403: AAA/SG/TRANSAC: Estimated Outstanding Transaction: 6, Current Max
Transaction: 6
```

The table below describes the significant fields shown in the display.

***Table 4: debug aaa dead-criteria transaction Field Descriptions***

| Field | Description |
|---|---|
| AAA/SG/TRANSAC | AAA server-group transaction. |
| Computed Retransmit Tries | Currently computed number of retransmissions before the server is marked as dead. |
| Current Tries | Number of successive failures since the last valid response. |
| Current Max Tries | Maximum number of tries since the last successful transaction. |

| Field | Description |
|---|---|
| Computed Dead Detect Interval | Period of inactivity (the number of seconds since the last successful transaction) that can elapse before the server is marked as dead. The period of inactivity starts when a transaction is sent to a server that is considered live. The dead-detect interval is the period that the device waits for responses from the server before the device marks the server as dead. |
| Elapsed Time | Amount of time that has elapsed since the last valid response. |
| Current Max Interval | Maximum period of inactivity since the last successful transaction. |
| Estimated Outstanding Transaction | Estimated number of transaction that are associated with the server. |
| Current Max Transaction | Maximum transaction since the last successful transaction. |

## Related Commands

| Command | Description |
|---|---|
| **radius-server dead-criteria** | Forces one or both of the criteria, used to mark a RADIUS server as dead, to be the indicated constant. |
| **show aaa dead-criteria** | Displays dead-criteria detection information for an AAA server. |

# debug umbrella

To enable debugging of the Cisco Umbrella Integration feature, use the **debug umbrella** command in privileged EXEC mode. To disable debugging, use the **no** form of this command.

**debug umbrella** {**config** | **device-registration** | **dnscrypt** | **redundancy**}
**no debug umbrella** {**config** | **device-registration** | **dnscrypt** | **redundancy**}

## Syntax Description

| | |
|---|---|
| **config** | Enables configuration debugging. |
| **device-registration** | Enables device registration debugging. |
| **dnscrypt** | Enables DNSCrypt debugging. |
| **redundancy** | Enables redundancy debugging. |

## Command Default

Debugging is disabled.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Examples

The following example shows how to enable Cisco Umbrella configuration debugging:

```
Device> enable
Device# debug umbrella config

Umbrella config debugging is on

Device# configure terminal
Device(config)# interface gigabitethernet 1/0/12
Device(config-if)# umbrella in test

*Nov 27 08:34:41.088: UMBRELLA-CONFIG:Umbrella token configured, so set mode as TOKEN
*Nov 27 08:34:41.088: UMBRELLA-CONFIG:check user configured resolver count
*Nov 27 08:34:41.088: UMBRELLA-CONFIG:Umbrella interface with no direct cloud access
*Nov 27 08:34:41.088: UMBRELLA-CONFIG:Umbrella mandatory parameter 'token' or
'api-key/secret/orgid' configured
*Nov 27 08:34:41.088: UMBRELLA-CONFIG:Processing is umbrella enabled check
*Nov 27 08:34:41.088: UMBRELLA-CONFIG:Is umbrella enabled check failed:sw idb info not found
*Nov 27 08:34:41.088: UMBRELLA-CONFIG:Send the interface info to device registration proces
*Nov 27 08:34:41.089: UMBRELLA-CONFIG:Add interface GigabitEthernet1/0/12 request sent to
DP
*Nov 27 08:34:41.089: UMBRELLA-CONFIG:Configured 'umbrella in test' on interface
GigabitEthernet1/0/12
*Nov 27 08:34:41.089: UMBRELLA-CONFIG:Cannot add domain patterns to DSA: Nothing to add
```

# delay-protection

To configure MKA to use delay protection in sending MACsec Key Agreement Protocol Data Units (MKPDUs), use the **delay-protection** command in MKA-policy configuration mode. To disable delay protection, use the **no** form of this command.

**delay-protection**
**no delay-protection**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Delay protection for sending MKPDUs is disabled.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Examples

The following example shows how to configure MKA to use delay protection in sending MKPDUs:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# delay-protection
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **confidentiality-offset** | Sets the confidentiality offset for MACsec operations. |
| **include-icv-indicator** | Includes ICV indicator in MKPDU. |
| **key-server** | Configures MKA key-server options. |
| **macsec-cipher-suite** | Configures cipher suite for deriving SAK. |
| **sak-rekey** | Configures the SAK rekey interval. |
| **send-secure-announcements** | Configures MKA to send secure announcements in sending MKPDUs. |
| **ssci-based-on-sci** | Computes SSCI based on the SCI. |
| **use-updated-eth-header** | Uses the updated Ethernet header for ICV calculation. |

# deny (MAC access-list configuration)

To prevent non-IP traffic from being forwarded if the conditions are matched, use the **deny** command in MAC access-list extended configuration mode. To remove a deny condition from the named MAC access list, use the **no** form of this command.

**deny** {**any** | **host** *src-MAC-addr* | *src-MAC-addr mask*} {**any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask* | **aarp** | **amber** | **appletalk** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **lsap** *lsap mask* | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp**] [**cos** *cos*]
**no deny** {**any** | **host** *src-MAC-addr* | *src-MAC-addr mask*} {**any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask* | **aarp** | **amber** | **appletalk** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **lsap** *lsap mask* | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp**] [**cos** *cos*]

## Syntax Description

| | |
|---|---|
| **any** | Denies any source or destination MAC address. |
| **host** *src-MAC-addr* \| *src-MAC-addr mask* | Defines a host MAC address and optional subnet matches the defined address, non-IP traffic from |
| **host** *dst-MAC-addr* \| *dst-MAC-addr mask* | Defines a destination MAC address and optional a packet matches the defined address, non-IP traf |
| *type mask* | (Optional) Specifies the EtherType number of a pa to identify the protocol of the packet.<br>The type is 0 to 65535, specified in hexadecimal.<br>The mask is a mask of don't care bits applied to t |
| **aarp** | (Optional) Specifies EtherType AppleTalk Addres address to a network address. |
| **amber** | (Optional) Specifies EtherType DEC-Amber. |
| **appletalk** | (Optional) Specifies EtherType AppleTalk/Ether |
| **dec-spanning** | (Optional) Specifies EtherType Digital Equipmen |
| **decnet-iv** | (Optional) Specifies EtherType DECnet Phase IV |
| **diagnostic** | (Optional) Specifies EtherType DEC-Diagnostic. |
| **dsm** | (Optional) Specifies EtherType DEC-DSM. |
| **etype-6000** | (Optional) Specifies EtherType 0x6000. |
| **etype-8042** | (Optional) Specifies EtherType 0x8042. |
| **lat** | (Optional) Specifies EtherType DEC-LAT. |
| **lavc-sca** | (Optional) Specifies EtherType DEC-LAVC-SCA |

| | |
|---|---|
| **lsap** *lsap-number mask* | (Optional) Specifies the LSAP number (0 to 6 identify the protocol of the packet.<br>*mask* is a mask of don't care bits applied to th |
| **mop-console** | (Optional) Specifies EtherType DEC-MOP R |
| **mop-dump** | (Optional) Specifies EtherType DEC-MOP D |
| **msdos** | (Optional) Specifies EtherType DEC-MSDOS |
| **mumps** | (Optional) Specifies EtherType DEC-MUMP |
| **netbios** | (Optional) Specifies EtherType DEC- Networ |
| **vines-echo** | (Optional) Specifies EtherType Virtual Integra Banyan Systems. |
| **vines-ip** | (Optional) Specifies EtherType VINES IP. |
| **xns-idp** | (Optional) Specifies EtherType Xerox Netwo an arbitrary EtherType in decimal, hexadecim |
| **cos** *cos* | (Optional) Specifies a class of service (CoS) n CoS can be performed only in hardware. A war is configured. |

## Command Default

This command has no defaults. However, the default action for a MAC-named ACL is to deny.

## Command Modes

MAC-access list extended configuration (config-ext-macl)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

You enter MAC-access list extended configuration mode by using the **mac access-list extended** global configuration command.

If you use the **host** keyword, you cannot enter an address mask; if you do not use the **host** keyword, you must enter an address mask.

When an access control entry (ACE) is added to an access control list, an implied **deny-any-any** condition exists at the end of the list. That is, if there are no matches, the packets are denied. However, before the first ACE is added, the list permits all packets.

To filter IPX traffic, you use the *type mask* or **lsap** *lsap mask* keywords, depending on the type of IPX encapsulation being used. Filter criteria for IPX encapsulation types as specified in Novell terminology and Cisco IOS XE terminology are listed in the table.

*Table 5: IPX Filtering Criteria*

| IPX Encapsulation Type | | Filter Criterion |
|---|---|---|
| **Cisco IOS XE Name** | **Novel Name** | |
| arpa | Ethernet II | EtherType 0x8137 |
| snap | Ethernet-snap | EtherType 0x8137 |
| sap | Ethernet 802.2 | LSAP 0xE0E0 |
| novell-ether | Ethernet 802.3 | LSAP 0xFFFF |

This example shows how to define the named MAC extended access list to deny NETBIOS traffic from any source to MAC address 00c0.00a0.03fa. Traffic matching this list is denied.

```
Device> enable
Device# configure terminal
Device(config)# mac access-list extended mac_layer
Device(config-ext-macl)# deny any host 00c0.00a0.03fa netbios.
Device(config-ext-macl)# end
```

This example shows how to remove the deny condition from the named MAC extended access list:

```
Device> enable
Device# configure terminal
Device(config)# mac access-list extended mac_layer
Device(config-ext-macl)# no deny any 00c0.00a0.03fa 0000.0000.0000 netbios.
Device(config-ext-macl)# end
```

The following example shows how to deny all packets with EtherType 0x4321:

```
Device> enable
Device# configure terminal
Device(config)# mac access-list extended mac_layer
Device(config-ext-macl)# deny any any 0x4321 0
Device(config-ext-macl)# end
```

You can verify your settings by entering the **show access-lists** privileged EXEC command.

## Related Commands

| Command | Description |
|---|---|
| **mac access-list extended** | Creates an access list based on MAC addresses for |
| **permit** | Permits from the MAC access-list configuration.<br>Permits non-IP traffic to be forwarded if conditions |
| **show access-lists** | Displays access control lists configured on a device |

# device-role (IPv6 snooping)

To specify the role of the device attached to the port, use the **device-role** command in IPv6 snooping configuration mode. To remove the specification, use the **no** form of this command.

**device-role** {**node** | **switch**}
**no device-role** {**node** | **switch**}

## Syntax Description

| | |
|---|---|
| **node** | Sets the role of the attached device to node. |
| **switch** | Sets the role of the attached device to device. |

## Command Default

The device role is node.

## Command Modes

IPv6 snooping configuration (config-ipv6-snooping)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **device-role** command specifies the role of the device attached to the port. By default, the device role is node.

The **switch** keyword indicates that the remote device is a switch and that the local switch is now operating in multiswitch mode; binding entries learned from the port will be marked with trunk_port preference level. If the port is configured as a trust-port, binding entries will be marked with trunk_trusted_port preference level.

This example shows how to define an IPv6 snooping policy name as policy1, place the device in IPv6 snooping configuration mode, and configure the device as the node:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# device-role node
Device(config-ipv6-snooping)# end
```

# device-role (IPv6 nd inspection)

To specify the role of the device attached to the port, use the **device-role** command in neighbor discovery (ND) inspection policy configuration mode.

**device-role** {**host** | **switch**}

## Syntax Description

| | |
|---|---|
| **host** | Sets the role of the attached device to host. |
| **switch** | Sets the role of the attached device to switch. |

## Command Default

The device role is host.

## Command Modes

ND inspection policy configuration (config-nd-inspection)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **device-role** command specifies the role of the device attached to the port. By default, the device role is host, and therefore all the inbound router advertisement and redirect messages are blocked.

The **switch** keyword indicates that the remote device is a switch and that the local switch is now operating in multiswitch mode; binding entries learned from the port will be marked with trunk_port preference level. If the port is configured as a trust-port, binding entries will be marked with trunk_trusted_port preference level.

The following example defines a Neighbor Discovery Protocol (NDP) policy name as policy1, places the device in ND inspection policy configuration mode, and configures the device as the host:

```
Device> enable
Device# configure terminal
Device(config)#  ipv6 nd inspection policy policy1
Device(config-nd-inspection)# device-role host
Device(config-nd-inspection)# end
```

# device-tracking (interface config)

To enable SISF-based device tracking and attach the *default* policy to an interface or VLAN, or to enable the feature and attach a custom policy enter the **device-tracking** command in interface configuration mode. To detach the policy from the interface or VLAN and revert to default, use the **no** form of the command.

**device-tracking** [ **attach-policy** *policy-name* ] [ **vlan** { *vlan-id* | **add** *vlan-id* | **all** | **except** *vlan-id* | **none** | **remove** *vlan-id* } ]
**no device-tracking** [ **attach-policy** *policy-name* ] [ **vlan** { *vlan-id* | **add** *vlan-id* | **all** | **except** *vlan-id* | **none** | **remove** *vlan-id* } ]

## Syntax Description

| | |
|---|---|
| **attach-policy** *policy-name* | Attaches the custom policy that you specify, to the interface and all VLANs. |
| **vlan** { *vlan-id* \| **add** *vlan-id* \| **all** \| **except** *vlan-id* \| **none** \| **remove** *vlan-id* } | Configures the VLAN list for the policy and attaches the custom policy to the specified VLANs. You can specify the following particulars:<br>• *vlan-id*: Enter one or more VLAN IDs. The custom policy is attached to all the VLAN IDs.<br>• **add***vlan-id*: Adds specified VLANs to the existing list of VLAN IDs. The custom policy is attached to all the VLAN IDs.<br>• **all**: Attaches the custom policy to all VLAN IDs.<br>This is the default option.<br>• **except***vlan-id*: Attaches the custom policy to all VLAN IDs, except the ones you specify here.<br>• **none**: Does not attach the custom policy to any VLAN.<br>**remove***vlan-id*: Removes specified VLANs from the existing list of VLAN IDs. The custom policy is attached only to the VLAN IDs in the list. |

## Command Default

SISF-based device tracking is disabled and a policy is not attached to the interface.

## Command Modes

Interface configuration [Device((config-if)# )]

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

If you enter the **device-tracking** command in the interface configuration mode, without any other keywords, the system attaches the *default* policy the interface or VLAN. The default policy is a built-in policy with default settings; you cannot change any of the attributes of the default policy.

If you configure the **device-tracking attach-policy***policy-name* command in the interface configuration mode, you can specify a custom policy name. You must have created the custom policy in global configuration mode already. The policy is attached to the specifed interface. You can then also specify the VLANs that you want to attach it to.

If you want to change the custom policy that is attached to a target, reconfigure the **device-tracking attach-policy***policy-name* command.

If you want to disable the feature on a particular target, enter the **no device-tracking** command in the interface configuration mode.

### Examples


- Example: Enabling SISF-Based Device Tracking and Attaching the Default Policy, on page 64
- Attaching a Custom Policy, on page 64
- Example: Disabling SISF-Based Device-Tracking , on page 65


## Examples

The following example shows how to enable SISF-based device tracking and attach the default policy to an interface. The default policy has default policy parameters, none of which can be changed:

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# interface tengigabitethernet1/0/1
Device(config-if)# device-tracking
Device(config-if)# end

Device# show device-tracking policies detail
Target               Type  Policy               Feature        Target range
Te1/0/1              PORT  default              Device-tracking vlan all
Te1/0/2              PORT  default              Device-tracking vlan all

Device-tracking policy default configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
Policy default is applied on the following targets:
Target               Type  Policy               Feature        Target range
Te1/0/1              PORT  default              Device-tracking vlan all
Te1/0/2              PORT  default              Device-tracking vlan all
```

## Examples

The following example shows how enable SISF-based device tracking and attach a custom policy called `sisf-01`, to the same interface as the above example, that is, `Te1/0/1`. Doing so replaces the existing default policy with custom policy `sisf-01` on `Te1/0/1`.

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# interface tengigabitethernet1/0/1
Device(config-if)# device-tracking attach-policy sisf-01
Device(config-if)# end

Device# show device-tracking policies detail
Target               Type  Policy               Feature        Target range
Te1/0/1              PORT  sisf-01              Device-tracking vlan all
Te1/0/2              PORT  default              Device-tracking vlan all

Device-tracking policy default configuration:
```

```
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
Policy default is applied on the following targets:
Target              Type  Policy              Feature        Target range
Te1/0/2             PORT  default             Device-tracking vlan all
 Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count 3000
Policy sisf-01 is applied on the following targets:
Target              Type  Policy              Feature        Target range
Te1/0/1             PORT  sisf-01             Device-tracking vlan all
```

## Examples

The following example shows how to disable SISF-based device-tracking on a target. The feature is disabled on target Te1/0/1. This is the same interface where a custom policy is applied in the previous example. The default policy continues to be available on the other interface where the feature is enabled, that is, `Te1/0/2`.

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# interface tengigabitethernet1/0/1
Device(config-if)# no device-tracking attach-policy sisf-01
Device(config-if)# end

Device# show device-tracking policies detail
Target              Type  Policy              Feature        Target range
Te1/0/2             PORT  default             Device-tracking vlan all

Device-tracking policy default configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
Policy default is applied on the following targets:
Target              Type  Policy              Feature        Target range
Te1/0/2             PORT  default             Device-tracking vlan all
```

# device-tracking (VLAN config)

To enable Switch Integrated Security Features (SISF)-based device tracking and attach the *default* policy to a VLAN, or to enable the feature, attach a custom policy to a VLAN, and specify policy priority, enter the **device-tracking** command in VLAN configuration mode. To detach the policy from a VLAN and revert to default, use the **no** form of the command.

**device-tracking** [ **attach-policy** *policy-name* ] [ **priority** *priority-value* ]

## Syntax Description

| | |
|---|---|
| **attach-policy** *policy-name* | Attaches the custom policy that you specify, to the VLAN. |
| **priority** *priority-value* | **Note** Although visible on the CLI, configuring this command has no effect. Policy priority is system-determined. You cannot change this. |

## Command Default

SISF-based device tracking is disabled.

## Command Modes

VLAN configuration mode [Device((config-vlan-config)# )]

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced |

## Usage Guidelines

If you enter the **device-tracking** command in VLAN configuration mode, without any other keywords, the system attaches the *default* policy to the VLAN. The default policy is a built-in policy with default settings; you cannot change any of the parameters of the default policy.

If you configure the **device-tracking attach-policy***policy-name* command in VLAN configuration mode, the custom policy you specify is attached to the VLAN. With a custom policy, you can configure certain parameters of a custom policy.

You can enable the feature and attach a policy - custom or default - to one or more VLANs or a range of VLANs.

### Examples


- Example: Enabling SISF-Based Device Tracking and Attaching the Default Policy, on page 66
- Example: Attaching a Custom Policy to a VLAN, on page 67
- Example: Attaching a Custom Policy to a Range of VLANs, on page 67


## Examples

The following example shows how to enable SISF-based device tracking and attach the default policy to VLAN 500:

```
Device# show device-tracking policies
Target               Type  Policy              Feature        Target range
Te1/0/1              PORT  sisf-03             Device-tracking vlan all
```

```
Te1/0/1             PORT  default             Address Resolution Relay vlan all
Te1/0/2             PORT  default             Device-tracking vlan all
vlan 333            VLAN  sisf-01             Device-tracking vlan all


Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)#vlan configuration 500
Device(config-vlan-config)# device-tracking
Device(config-vlan-config)# end


Device#show device-tracking policies
Target              Type  Policy              Feature        Target range
Te1/0/1             PORT  sisf-03             Device-tracking vlan all
Te1/0/1             PORT  default             Address Resolution Relay vlan all
Te1/0/2             PORT  default             Device-tracking vlan all
vlan 333            VLAN  sisf-01             Device-tracking vlan allvlan 500
   VLAN  default              Device-tracking vlan all
```

## Examples

The following example shows how to attach a custom policy called sisf-03, to the same VLAN as the above example, that is, VLAN 500. Doing so replaces the existing default policy with custom policy sisf-03 on the VLAN:

```
Device# show device-tracking policies
Target              Type  Policy              Feature        Target range
Te1/0/1             PORT  sisf-03             Device-tracking vlan all
Te1/0/1             PORT  default             Address Resolution Relay vlan all
Te1/0/2             PORT  default             Device-tracking vlan all
vlan 333            VLAN  sisf-01             Device-tracking vlan all
vlan 500            VLAN  default             Device-tracking vlan all

Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# vlan configuration 500
Device(config-vlan-config)# device-tracking attach-policy sisf-03
Device(config-vlan-config)# end

Device# show device-tracking policies
Target              Type  Policy              Feature        Target range
Te1/0/1             PORT  sisf-03             Device-tracking vlan all
Te1/0/1             PORT  default             Address Resolution Relay vlan all
Te1/0/2             PORT  default             Device-tracking vlan all
vlan 333            VLAN  sisf-01             Device-tracking vlan allvlan 500
   VLAN  sisf-03              Device-tracking vlan all
```

## Examples

The following example shows how to attach a custom policy to a range of VLANs (VLANs 10 to 15):

```
Device(config)# vlan configuration 10-15
Device(config-vlan-config)#device-tracking attach-policy sisf-01
Device(config-vlan-config)#end

Device# show device-tracking policies
Target              Type  Policy              Feature        Target range
Te1/0/2             PORT  default             Device-tracking vlan all
vlan 10             VLAN  sisf-01             Device-tracking vlan all
vlan 11             VLAN  sisf-01             Device-tracking vlan all
vlan 12             VLAN  sisf-01             Device-tracking vlan all
vlan 13             VLAN  sisf-01             Device-tracking vlan all
```

```
vlan 14             VLAN  sisf-01             Device-tracking vlan all
vlan 15             VLAN  sisf-01             Device-tracking vlan all
```

# device-tracking binding

To specify how binding entries are maintained in the binding table, enter the **device-tracking binding** command in global configuration mode. With this command you can configure the lifetime of each state, the maximum number of entries allowed in a binding table, and whether binding entry events are logged. You can also use this command to configure static binding entries. To revert to the default value, use the **no** form of the command.

**device-tracking binding** { **down-lifetime** | **logging** | **max-entries** | **reachable-lifetime** | **stale-lifetime** | **vlan** }

For the sake of clarity, the remaining command string after each one of the above options is listed separately:

- **device-tracking binding down-lifetime** { *seconds* | **infinite** }

  **no device-tracking binding down-lifetime**

- **device-tracking binding logging**

  **no device-tracking binding logging**

- **device-tracking binding max-entries** *no_of_entries* [ **mac-limit** *no_of_entries* | **port-limit** *no_of_entries* [ **mac-limit** *no_of_entries* ] | **vlan-limit** *no_of_entries* [ **mac-limit** *no_of_entries* | **port-limit** *no_of_entries* [ **mac-limit** *no_of_entries* ] ] ]

  **no device-tracking binding max-entries**

- **device-tracking binding reachable-lifetime** { *seconds* | **infinite** } [ **down-lifetime** { *seconds* | **infinite** } | **stale-lifetime** { *seconds* | **infinite** } [ **down-lifetime** { *seconds* | **infinite** } ] ]

  **no device-tracking binding reachable-lifetime**

- **device-tracking binding stale-lifetime** { *seconds* | **infinite** } [ **down-lifetime** { *seconds* | **infinite** } ]

  **no device-tracking binding stale-lifetime**

- **device-tracking binding vlan** *vlan_id* { *ipv4_add ipv6_add ipv6_prefix* } [ **interface** *inteface_type_no* ] [ *48-bit-hardware-address* ] [ **reachable-lifetime** { *seconds* | **default** | **infinite** } | **tracking** { **default** | **disable** | **enable** [ **retry-interval** { *seconds* | **default** } ] } [ **reachable-lifetime** { *seconds* | **default** | **infinite** } ] ]

## Syntax Description

| | |
|---|---|
| **down-lifetime** { *seconds* \| **infinite** } | Provides the option to configure a countdown timer for a binding entry in the DOWN state, or, to disable the timer.<br><br>A binding entry enters the DOWN state when the host's connecting interface is administratively down. If a timer is configured, one of these events may occur before timer expiry - either the interface can be up again, or, the entry can *remain* in the DOWN state. If the interface is up before timer expiry, the timer is stopped, and the state of the entry changes. If the entry remains in the DOWN state after timer expiry, it is removed from the binding table. If the timer is disabled or turned off, the entry is never removed from the binding table and can remain in the DOWN state indefinitely, or until the interface is up again.<br><br>Configure one of these options:<br><br>• *seconds*: Configure a value for the down-lifetime timer. Enter a value between 1 and 86400 seconds. The default value is 86400 seconds (24 hours).<br><br>• **infinite**: Disables the timer for the DOWN state. This means that a timer is not started when an entry enters the DOWN state. |
| **logging** | Enables generation of logs for binding entry events. |
| **device-tracking binding max-entries** *no_of_entries* [ **mac-limit** *no_of_entries* \| **port-limit** *no_of_entries* \| **vlan-limit** *no_of_entries* ] | Configures the maximum number of entries for a binding table. Enter a value between 1 and 200000. The default value is 200000.<br><br>**Note**<br>This limit applies only to dynamic entries and not static binding entries.<br><br>Optionally, you can also configure these limits:<br><br>• **mac-limit** *no_of_entries*: Configures the maximum number of entries allowed per MAC address. Enter a value between 1 and 100000. By default, a limit is not set.<br><br>• **port-limit** *no_of_entries*Configures the maximum number of entries allowed per interface. Enter a value between 1 and 100000. By default, a limit is not set.<br><br>• **vlan-limit** *no_of_entries*: Configures the maximum number of entries allowed per VLAN. Enter a value between 1 and 100000. By default, a limit is not set.<br><br>The **no** form of the command resets the **max-entries** value to 200000 and sets the **mac-limit**, **port-limit**, **vlan-limit** to "no limit". |

| | |
|---|---|
| **reachable-lifetime** { *seconds* \| **infinite** } | Provides the option to configure a countdown timer for a binding entry in the REACHABLE state, or, to disable the timer.<br><br>If a timer is configured, either one of these events may occur before timer expiry - incoming packets are received from the host, or there are no incoming packets from the host. Every time an incoming packet is received from the host, the timer is reset. If no incoming packets are received and the timer expires, then the state of the entry changes based on the reachability of the host. If the timer is disabled or turned off, the entry can remain in the REACHABLE state, indefinitely.<br><br>Configure one of these options:<br><br>• *seconds*: Configure a value for the reachable-lifetime timer. Enter a value between 1 and 86400 seconds. The default value is 300 seconds (5 minutes).<br><br>• **infinite**: Disables the timer for the REACHABLE state. This means that a timer is not started when an entry enters the REACHABLE state. |
| **stale-lifetime** { *seconds* \| **infinite** } | Provides the option to configure a countdown timer for a binding entry in the STALE state, or, to disable the timer.<br><br>If a timer is configured, either one of these events may occur before timer expiry - incoming packets are received from the host, or there are no incoming packets from the host. If an incoming packet is received, the timer is stopped and the entry transitions to a new state. If no incoming packets are received and the timer expires, then the entry is removed from the binding table. If the timer is disabled or turned off, the entry can remain in the STALE state, indefinitely.<br><br>If polling is enabled, a final attempt is made to probe the host at stale timer expiry.<br><br>**Note**<br>If polling is enabled, polling occurs when the reachable lifetime timer expires (3 times), and then a final attempt at stale timer expiry as well. The time required to poll an entry after expiry of reachable lifetime, is subtracted from the stale lifetime.<br><br>Configure one of these options:<br><br>• *seconds*: Configure a value for the stale-lifetime timer. Enter a value between 1 and 86400 seconds. The default value is 86400 seconds (24 hours).<br><br>• **infinite**: Disables the timer for the STALE state. This means that a timer is not started when an entry enters the STALE state. |

| | |
|---|---|
| **device-tracking binding vlan** *vlan_id* { *ipv4_add ipv6_add ipv6_prefix* } [ **interface** *inteface_type_no* ] [ *48-bit-hardware-address* ] [ **reachable-lifetime** { *seconds* \| **default** \| **infinite** } \| **tracking** { **default** \| **disable** \| **enable** [ **retry-interval** { *seconds* \| **default** } ] } [ **reachable-lifetime** { *seconds* \| **default** \| **infinite** } ] ] | Creates a static binding entry in the binding table. You can also specify how static binding entries are maintained in the binding table.<br>**Note**<br>The limit you configure for the **max-entries** *no_of_entries* option (above) does not apply to static binding entires. There is no limit to the number of static entries you can create.<br>• Enter an IP address or prefix:<br>  • *ipv4_add* : Enter an IPv4 address.<br>  • *ipv6_add* : Enter an IPv6 address.<br>  • *ipv6_prefix* : Enter an IPv6 prefix.<br>• **interface** *inteface_type_no*: Enter an interface type and number. Use the question mark (?) online help function to display the types of interfaces available on the device.<br>• (Optional) *48-bit-hardware-address*: Enter a MAC address. If you do not configure a MAC address for the binding entry, any MAC address is allowed.<br>• (Optional) **reachable-lifetime** {*seconds* \| **default** \| **infinite** }: Configures the reachable lifetime settings for a static binding entry in the REACHABLE state. If you want to configure a reachable lifetime for a static binding entry, you must specify the MAC address for the entry.<br>If you do not configure a value, the same value as configured for **device-tracking binding reachable-lifetime** applies.<br>*seconds*: Configure a value for the reachable-lifetime timer. Enter a value between 1 and 86400 seconds. The default value is 300 seconds (5 minutes).<br>**default**: Uses the same value as configured for dynamic entries in the binding table.<br>**infinite**: Disables the timer for the REACHABLE state. This means that a timer is not started when a static binding entry enters the REACHABLE state.<br>• (Optional) **tracking** {**default** \| **disable** \| **enable**}: Configures polling related settings for a static binding entry.<br>**default**:Polling is disabled.<br>**disable**: Disables polling for a static binding entry.<br>**enable**: Enables polling for a static binding entry.<br>When you enable tracking, you also have the option to configure a **retry-interval**. This is a multiplicative factor or "base value", for the backoff algorithm. The backoff algorithm determines the wait time between the 3 polling attempts that occur after reachable lifetime expiry.<br>Enter a value between 1 and 3600 seconds. The default value is one. |

## Command Default

If you do not configure a value, the default values for down, reachable, and stale lifetimes, and maximum number of binding entries allowed in a binding table are applicable - as long as a policy-level value is not set. See the *Usage Guidelines* below for further details.

## Command Modes

Global configuration [Device(config)# ]

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **device-tracking binding** command enables you to specify how entries are maintained in a binding table, at a global level. The settings therefore apply to all interfaces and VLANs where SISF-based device-tracking is enabled. But for the system to start extracting binding information from packets that enter the network and to create binding entries to which the settings you configure here will apply, there must exist a policy that is attached an interface or VLAN.

If there is no policy on any interface or VLAN, the only entries that can exist in a binding table are any static binding entries you create.

### Changing Any Binding Entry Setting

When you reconfigure a value or setting with the **device-tracking binding** command, the change applies only to subsequently created binding entries. The changed configuration does not apply to existing entries. The older setting applies to an older entry.

To display the current settings, enter the **show device-tracking database** command in privileged EXEC mode.

### Global versus Policy-Level Settings

For some of the settings you configure with this command, there are policy level counterparts. (A policy level paramter is configured in the device-tracking configuration mode and applies only to that policy). The tables below clarifies when a globally configured value takes precedence and when a policy-level value takes precedence:

| **Option under device-tracking binding global configuration command** | **Policy-level counterpart in the device-tracking configuration mode** |
|---|---|
| **device-tracking binding reachable-lifetime** { *seconds* \| **infinite** } | **tracking enable** [**reachable-lifetime** [*seconds* \| **infinite**] ] |
| `Device(config)# device-tracking binding reachable-lifetime 2000` | `Device(config)# device-tracking policy sisf-01`<br>`Device(config-device-tracking)#`<br>`Device(config-device-tracking)# tracking enable`<br><br>`reachable-lifetime 250` |
| If a policy-level value *and* a globally configured value exists, the policy-level value applies.<br>If only a globally configured value exists, the globally configured value applies.<br>If only a policy-level value exists the policy-level value applies.<br>See: Example: Configuring a Reachable, Stale, and Down Lifetime at the Global vs Policy Level, on page 77. | |

| Option under device-tracking binding global configuration command | Policy-level counterpart in the device-tracking configuration mode |
|---|---|
| **device-tracking binding stale-lifetime** { *seconds* \| **infinite** } | **tracking disable** [**stale-lifetime** [*seconds* \| **infinite**] ] |
| `Device(config)# device-tracking binding stale-lifetime 2000` | `Device(config)# device-tracking policy sisf-01`<br>`Device(config-device-tracking)#`<br>`Device(config-device-tracking)# tracking enable`<br><br>`stale-lifetime 500` |
| If a policy-level value *and* a globally configured value exists, the policy-level value applies.<br>If only a globally configured value exists, the globally configured value applies.<br>If only a policy-level value exists the policy-level value applies.<br>See: Example: Configuring a Reachable, Stale, and Down Lifetime at the Global vs Policy Level, on page 77. | |

| Option under device-tracking binding global configuration command | Policy-level counterpart in the device-tracking configuration mode |
|---|---|
| **device-tracking binding max-entries** *no_of_entries* [ **mac-limit** *no_of_entries* \| **port-limit** *no_of_entries* \| **vlan-limit** *no_of_entries* ] | **limit address-count***ip-per-port* |
| `Device(config)# device-tracking binding max-entries 30 vlan-limit 25 port-limit 20 mac-limit 19` | `Device(config)# device-tracking policy sisf-01`<br>`Device(config-device-tracking)#`<br>`Device(config-device-tracking)# limit address-count 30` |
| If a policy-level value *and* globally configured values exist, the creation of binding entries is stopped when a limit is reached - this can be one of the global values or the policy-level value.<br>If only globally configured values exist, the creation of binding entries is stopped when a limit is reached.<br>If only a policy-level value exists, the creation of binding entries is stopped when the policy-level limit is reached.<br>Example: Global vs Policy-Level Address Limits, on page 81. | |

| Option under device-tracking binding global configuration command | Policy-level counterpart in the device-tracking configuration mode |
|---|---|
| **device-tracking binding max-entries** *no_of_entries* [ **mac-limit** *no_of_entries* ] | IPv4 per MAC and IPv6 per MAC<br>While you cannot configure either one of the above limits in a policy, a programmatically created policy may have either one, both, or neither one of the limits. |

| Option under device-tracking binding global configuration command | Policy-level counterpart in the device-tracking configuration mode |
|---|---|
| `Device(config)# device-tracking binding max-entries 300`<br>`mac-limit 3` | `Device# show device-tracking policy LISP-DT-GLEAN-VLAN`<br><br>`Policy LISP-DT-GLEAN-VLAN configuration:`<br>`  security-level glean (*)`<br>`  device-role node`<br>`  gleaning from Neighbor Discovery`<br>`  gleaning from DHCP`<br>`  gleaning from ARP`<br>`  gleaning from DHCP4`<br>`  NOT gleaning from protocol unkn`<br>`  limit address-count for IPv4 per mac 4 (*)`<br>`  limit address-count for IPv6 per mac 12 (*)`<br>`  tracking enable`<br>`<output truncated>` |
| If a policy-level value *and* globally configured values exists, the creation of binding entries is stopped when a limit is reached - this can be one of the global values or the policy-level value.<br><br>If only globally configured values exist, the creation of binding entries is stopped when a limit is reached.<br><br>If only a policy-level value exists, the creation of binding entries is stopped when the policy-level limit is reached. | |

### Configuring Down, Reachable, Stale Lifetimes

When you configure a non-default value for the **down-lifetime**, or **reachable-lifetime**, or **stale-lifetime** keywords, the system reverts the lifetimes that you do not configure, to default values. The following example clarifies this behaviour: Example: Configuring Non-Default Values for Reachable, Stale, and Down Lifetimes, on page 77.

To display the currently configured lifetime values, enter the **show running-config | include device-tracking** command in privileged EXEC mode.

### Configuring MAC, Port, VLAN Limits

When you configure a non-default value for the **mac-limit**, or **port-limit**, or **vlan-limit** keywords, the system reverts the limits that you do not configure, to default values.

To configure all three limits in the same command line, first configure the VLAN limit, then the port limit, and finally the MAC limit:

```
Device(config)# device-tracking binding max-entries 15 vlan-limit 2 port-limit 20 mac-limit
 5
```

You can also use this system behavior when you want to reset one or more - but not *all* limits, to their default values. Although the default for all three keywords is that there is no limit, you cannot enter the number "0" to set a limit to its default value. Zero is not within the valid value range for any of the limits. To reset one or more limits to their default values, leave out the corresponding keyword. The following example clarifies this behaviour: Example: Setting VLAN, Port, and MAC Limits to Default Values, on page 85.

### Enabling Logging of Binding Entry Events

When you configure the **device-tracking binding logging** global configuration command to generate logs for binding entry events, you may also have to configure a few general logging settings, depending on your requirements:

- (Required) The **logging buffered informational** command in global configuration mode.

With this command you enable message logging at a device level and you specify a severity level. Configuring the command allows logs to be copied and stored to a local, internal buffer. Specifying a severity level causes messages at that level and numerically lower levels to be logged.

Logs generated for binding entry events have a severity level of 6 (meaning, informational). For example:

```
%SISF-6-ENTRY_CREATED: Entry created IP=192.0.2.24 VLAN=200 MAC=001b.4411.4ab6 I/F=Te1/0/4
Preflevel=00FF
```

- (Optional) The **logging console** command in global configuration mode.

  With this command you send the logs to the console (all available TTY lines).

**Caution** A low severity level may cause the number of messages being displayed on the console to increase significantly. Further, the console is a slow display device. In message storms some logging messages may be silently dropped when the console queue becomes full. Set severity levels accordingly.

If you don't want to configure this command, you can view logs when required by entering the **show logging** command in privileged EXEC mode.

If the **logging console** command is not enabled, logs are not *displayed* on the device console, but if you have configured **device-tracking binding logging** and **logging buffered informational**, logs will be generated and available in the local buffer.

For information about the *kind* of binding entry events for which logs are generated, see the system message guide for the corresponding release: System Message Guides. Search for `SISF-6`.

While the **device-tracking binding logging** command logs binding entry events, there is also the **device-tracking logging** command, which enables snooping security logging. The two command log different kinds of events and the generated logs have different severity levels.

### Creating a Static Binding Entry

If there are silent but reachable hosts in the Layer 2 domain, and you want to retain binding information for these silent hosts, you can create static binding entries.

While there is no limit to the number of static entries you can create, these entries also contribute to the size of the binding table. Consider the number of such entries you require, before you create them.

You can create a static binding entry even if a policy is not attached to the interface or VLAN specified in the static binding entry.

When you configure a static binding entry followed by its settings (for example, reachable-lifetime), the configuration applies only to that static binding entry and not to any other entries, static or dynamic. The following example shows you how to created a static binding entry: Example: Creating a Static Binding Entry, on page 80.

## Examples

- Example: Configuring Non-Default Values for Reachable, Stale, and Down Lifetimes, on page 77
- Example: Configuring a Reachable, Stale, and Down Lifetime at the Global vs Policy Level, on page 77

- Example: Creating a Static Binding Entry, on page 80
- Example: Global vs Policy-Level Address Limits, on page 81
- Example: Setting VLAN, Port, and MAC Limits to Default Values, on page 85
- Example: Global vs Policy-Level Limits Relating to MAC Addresses, on page 86


### Example: Configuring Non-Default Values for Reachable, Stale, and Down Lifetimes

The following example clarifies system behaviour when you configure values for reachable, stale, and down lifetimes seperately (the effect is not cumulative). It also show you how to configure values in a way that configuration is retained for all the lifetimes.

In the first step of this example only a reachable-lifetime is configured. This means the down-lifetime and stale lifetime are set to default, because the **stale-lifetime** and **down-lifetime** keywords have been left out:

```
Device(config)# device-tracking binding reachable-lifetime 700
Device(config)# exit
Device# show running-config | include device-tracking
device-tracking policy sisf-01
 device-tracking attach-policy sisf-01
 device-tracking attach-policy sisf-01 vlan 200device-tracking binding reachable-lifetime
700
device-tracking binding logging
```

In the next step of this example, a stale-lifetime of 1500 seconds and a down-lifetime of 1000 seconds is configured. With this, the reachable-lifetime configured in the previous step, is to default:

```
Device(config)# device-tracking binding stale-lifetime 1500 down-lifetime 1000
Device(config)# exit
Device# show running-config | include device-tracking
device-tracking policy sisf-01
 device-tracking attach-policy sisf-01
 device-tracking attach-policy sisf-01 vlan 200device-tracking binding stale-lifetime 1500
 down-lifetime 1000
device-tracking binding logging
```

In the next step of this example, reachable, down, and stale lifetimes of 700, 1000, and 200 respectively, are configured. With this, the value for the stale-lifetime is changed from 1500 seconds, to 1000 seconds. The down-lifetime is changed from 1000 to 200. The reachable-lifetime is configured as 700 seconds.

```
Device(config)# device-tracking binding reachable-lifetime 700 stale-lifetime 1000
down-lifetime 200
Device(config)# exit
Device# show running-config | include device-tracking
device-tracking policy sisf-01
 device-tracking attach-policy sisf-01
 device-tracking attach-policy sisf-01 vlan 200device-tracking binding reachable-lifetime
700 stale-lifetime 1000 down-lifetime 200
device-tracking binding logging
```

If any one of the lifetimes requires a change and the values for the other lifetimes must be retained, all three keywords must be reconfigured with the required values - everytime, and in the same command line.

### Example: Configuring a Reachable, Stale, and Down Lifetime at the Global vs Policy Level

The following example shows you how to configure the reachable, stale, and down lifetimes for binding entries, at a global level. This example also shows you how you can then override the global setting and

configure a different lifetime for entries learnt on a particular interface or VLAN, by configuring a policy-level setting.

In the first part of the example, the output of the **show device-tracking policy** *policy-name* command shows that a policy-level value is not set and the default binding table settings are applicable to the existing entries. After a reachable, stale, and down lifetime is configured with the **device-tracking binding** command in global configuration mode, the new values are effective and are applied only to the four new entries that are added to the table.

**Note** In the output of the **show device-tracking database** command, note the `Time left` column for the binding entries. There is minor difference in the reachable lifetime of each entry. This is a system-imposed jitter (+/- 5 percent of the configured value), to ensure that system performance is not affected when a large number of entries are added to the binding table. Binding entries go through their lifecycle in a staggered manner thus preventing points of congestion.

Current configuration, which shows that policy-level reachable lifetime is not configured. The binding table entries show that the current reachable lifetime is 500 seconds (time left + age):

```
Device# show device-tracking policy sisf-01
Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
Policy sisf-01 is applied on the following targets:
Target               Type  Policy               Feature        Target range
Te1/0/4              PORT  sisf-01              Device-tracking vlan 200

Device# show device-tracking database
Binding Table has 4 entries, 4 dynamic (limit 200000)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned

Network Layer Address                          Link Layer Address     Interface  vlan
prlvl      age        state      Time left       <<<<
ARP 192.0.9.9                                  000a.959d.6816         Te1/0/4    200
0064       40s        REACHABLE  466 s
ARP 192.0.9.8                                  000a.959d.6816         Te1/0/4    200
0064       40s        REACHABLE  472 s
ARP 192.0.9.7                                  000a.959d.6816         Te1/0/4    200
0064       40s        REACHABLE  470 s
ARP 192.0.9.6                                  000a.959d.6816         Te1/0/4    200
0064       40s        REACHABLE  469 s
```

Configuration of reachable, stale and down lifetime at the global level. New values apply only to binding entries created after this:

```
Device(config)# device-tracking binding reachable-lifetime 700 stale-lifetime 1000
down-lifetime 200

Device # show device-tracking database
Binding Table has 8 entries, 8 dynamic (limit 200000)
```

```
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned

Network Layer Address                          Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.13                                 000a.959d.6816         Te1/0/4    200
00C8       4s         REACHABLE  699 s           <<<< new global value applied
ARP 192.0.9.12                                 000a.959d.6816         Te1/0/4    200
00C8       4s         REACHABLE  719 s           <<<< new global value applied
ARP 192.0.9.11                                 000a.959d.6816         Te1/0/4    200
00C8       4s         REACHABLE  728 s           <<<< new global value applied
ARP 192.0.9.10                                 000a.959d.6816         Te1/0/4    200
00C8       4s         REACHABLE  712 s           <<<< new global value applied
ARP 192.0.9.9                                  000a.959d.6816         Te1/0/4    200
0064       9mn        STALE      try 0 1209 s
ARP 192.0.9.8                                  000a.959d.6816         Te1/0/4    200
0064       9mn        VERIFY     5 s try 3
ARP 192.0.9.7                                  000a.959d.6816         Te1/0/4    200
0064       9mn        VERIFY     2816 ms try 3
ARP 192.0.9.6                                  000a.959d.6816         Te1/0/4    200
0064       9mn        VERIFY     1792 ms try 3
```

In this second part of the example, a policy level value is configured and the reachable lifetime is set to 50 seconds. This new reachable lifetime is again applicable only to entries created after this.

Only a reachable lifetime is configured at the policy-level and not a stale and down lifetime. This means it is still the global values that apply if the reachable lifetime of the two new entries expires and they move to the STALE or DOWN state.

```
Device(config)# device-tracking policy sisf-01
Device(config-device-tracking)# tracking enable reachable-lifetime 50
Device# show device-tracking policy sisf-01
Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  tracking enable reachable-lifetime 50    <<<< new value applies only to binding entries
created after this and on interfaces and VLANs where this policy is attached.
Policy sisf-01 is applied on the following targets:
Target               Type  Policy               Feature        Target range
Te1/0/4              PORT  sisf-01              Device-tracking vlan 200

Device# show device-tracking database
Binding Table has 10 entries, 10 dynamic (limit 200000)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


Network Layer Address                          Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.21                                 000a.959d.6816         Te1/0/4    200
```

```
0064       5s         REACHABLE  45 s           <<<< new policy-level value applied
ARP 192.0.9.20                                 000a.959d.6816         Te1/0/4    200
0064       5s         REACHABLE  46 s           <<<< new policy-level value applied
ARP 192.0.9.13                                 000a.959d.6816         Te1/0/4    200
00C8       14mn       STALE     try 0 865 s
ARP 192.0.9.12                                 000a.959d.6816         Te1/0/4    200
00C8       14mn       STALE     try 0 183 s
ARP 192.0.9.11                                 000a.959d.6816         Te1/0/4    200
00C8       14mn       STALE     try 0 178 s
ARP 192.0.9.10                                 000a.959d.6816         Te1/0/4    200
00C8       14mn       STALE     try 0 165 s
ARP 192.0.9.9                                  000a.959d.6816         Te1/0/4    200
0064       23mn       STALE     try 0 327 s
ARP 192.0.9.8                                  000a.959d.6816         Te1/0/4    200
0064       23mn       STALE     try 0 286 s
ARP 192.0.9.7                                  000a.959d.6816         Te1/0/4    200
0064       23mn       STALE     try 0 303 s
ARP 192.0.9.6                                  000a.959d.6816         Te1/0/4    200
0064       23mn       STALE     try 0 306 s

Device# show device-tracking database <<<< checking binding table again after new policy-level
 reachable-lifetime expires
Binding Table has 7 entries, 7 dynamic (limit 200000)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned

Network Layer Address                          Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.21                                 000a.959d.6816         Te1/0/4    200
0064       3mn        STALE     try 0 887 s  <<<< global value applies for stale-lifetime;
  policy-level value was not configured
ARP 192.0.9.20                                 000a.959d.6816         Te1/0/4    200
0064       3mn        STALE     try 0 884 s  <<<< global value applies for stale-lifetime;
  policy-level value was not configured
ARP 192.0.9.13                                 000a.959d.6816         Te1/0/4    200
00C8       17mn       STALE     try 0 664 s
ARP 192.0.9.9                                  000a.959d.6816         Te1/0/4    200
0064       27mn       STALE     try 0 136 s
ARP 192.0.9.8                                  000a.959d.6816         Te1/0/4    200
0064       27mn       STALE     try 0 96 s
ARP 192.0.9.7                                  000a.959d.6816         Te1/0/4    200
0064       27mn       STALE     try 0 108 s
ARP 192.0.9.6                                  000a.959d.6816         Te1/0/4    200
0064       27mn       STALE     try 0 111 s
```

### Example: Creating a Static Binding Entry

The following example shows you how to create a static binding entry. The "S" at the beginning of the entry indicates that it is a static binding entry:

```
Device(config)# device-tracking binding vlan 100 192.0.2.1 interface tengigabitethernet1/0/1
 00:00:5e:00:53:af reachable-lifetime infinite
Device(config)# exit
Device# show device-tracking database
Binding Table has 2 entries, 0 dynamic (limit 200000)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
```

```
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


    Network Layer Address                   Link Layer Address     Interface  vlan
prlvl      age        state      Time left
S   192.0.2.1                               0000.5e00.53af         Te1/0/1    100
0100       14s        REACHABLE  N/A
```

### Example: Global vs Policy-Level Address Limits

The following example show you how to assess which address limit is reached, when you configure address limits at the global level and at the policy-level.

The global level settings refer to the values configured for the following command string: **device-tracking bindingmax-entries** *no_of_entries* [**mac-limit** *no_of_entries* | **port-limit** *no_of_entries* | **vlan-limit** *no_of_entries*]

The policy level parameter refers to the **limit address-count** option in the device-tracking configuration mode.

For this first part of the example, the configuration is as follows:

- Global configuration: max-entries=30, vlan-limit=25, port-limit=20, mac-limit=19.
- Policy-level configuration: limit address-count=45.

The output of the **show device-tracking database details** privileged EXEC command shows that the port limit (`max/port`) is reached first. A maximum of 20 entries are allowed on a port or interface. No further binding entries are created after this. While the mac limit is configured with a lower absolute value (19), the output of the **show device-tracking database mac** privileged EXEC command shows that there are only 3 unique MAC address in the list of binding entries in the table - this limit is therefore not reached.

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# device-tracking binding max-entries 30 vlan-limit 25 port-limit 20 mac-limit
 19
Device(config)# device-tracking policy sisf-01
Device(config-device-tracking)# limit address-count 45
Device(config-device-tracking)# end
Device# show device-tracking policy sisf-01
Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count 45
Policy sisf-01 is applied on the following targets:
Target              Type  Policy              Feature        Target range
Te1/0/4             PORT  sisf-01             Device-tracking vlan 200

Device# show device-tracking database details
 Binding table configuration:
 ---------------------------
 max/box  : 30
 max/vlan : 25
 max/port : 20
 max/mac  : 19

 Binding table current counters:
 ------------------------------
```

```
 dynamic  : 20
 local    : 0
 total    : 20    <<<< no further entries created after this.

 Binding table counters by state:
 ---------------------------------
 REACHABLE  : 20
   total    : 20
<output truncated>

Device# show device-tracking database
Binding Table has 20 entries, 20 dynamic (limit 30)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned

Network Layer Address                          Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.39                                 000c.959d.6816         Te1/0/4    200
0064       14s        REACHABLE  37 s
ARP 192.0.9.38                                 000b.959d.6816         Te1/0/4    200
0064       14s        REACHABLE  37 s
ARP 192.0.9.37                                 000b.959d.6816         Te1/0/4    200
0064       14s        REACHABLE  36 s
ARP 192.0.9.36                                 000b.959d.6816         Te1/0/4    200
0064       14s        REACHABLE  39 s
ARP 192.0.9.35                                 000b.959d.6816         Te1/0/4    200
0064       14s        REACHABLE  38 s
ARP 192.0.9.34                                 000b.959d.6816         Te1/0/4    200
0064       14s        REACHABLE  37 s
ARP 192.0.9.33                                 000b.959d.6816         Te1/0/4    200
0064       15s        REACHABLE  36 s
ARP 192.0.9.32                                 000b.959d.6816         Te1/0/4    200
0064       15s        REACHABLE  37 s
ARP 192.0.9.31                                 000b.959d.6816         Te1/0/4    200
0064       15s        REACHABLE  36 s
ARP 192.0.9.30                                 000b.959d.6816         Te1/0/4    200
0064       15s        REACHABLE  36 s
ARP 192.0.9.29                                 000b.959d.6816         Te1/0/4    200
0064       15s        REACHABLE  35 s
ARP 192.0.9.28                                 000a.959d.6816         Te1/0/4    200
0064       15s        REACHABLE  36 s
ARP 192.0.9.27                                 000a.959d.6816         Te1/0/4    200
0064       16s        REACHABLE  35 s
ARP 192.0.9.26                                 000a.959d.6816         Te1/0/4    200
0064       16s        REACHABLE  36 s
ARP 192.0.9.25                                 000a.959d.6816         Te1/0/4    200
0064       16s        REACHABLE  34 s
ARP 192.0.9.24                                 000a.959d.6816         Te1/0/4    200
0064       16s        REACHABLE  35 s
ARP 192.0.9.23                                 000a.959d.6816         Te1/0/4    200
0064       16s        REACHABLE  34 s
ARP 192.0.9.22                                 000a.959d.6816         Te1/0/4    200
0064       16s        REACHABLE  36 s
ARP 192.0.9.21                                 000a.959d.6816         Te1/0/4    200
0064       17s        REACHABLE  33 s
ARP 192.0.9.20                                 000a.959d.6816         Te1/0/4    200
0064       17s        REACHABLE  33 s

Device# show device-tracking database mac
 MAC                    Interface  vlan       prlvl      state            Time left
```

```
 Policy           Input_index
000c.959d.6816         Te1/0/4    200        NO TRUST   MAC-REACHABLE    27 s
sisf-01          12
000b.959d.6816         Te1/0/4    200        NO TRUST   MAC-REACHABLE    27 s
sisf-01          12
000a.959d.6816         Te1/0/4    200        NO TRUST   MAC-REACHABLE    27 s
sisf-01          12
```

For this second part of the example, the configuration is as follows:

- Global configuration: max-entries=30, vlan-limit=25, port-limit=20, mac-limit=19.
- Policy-level configuration: limit address-count=14.

The limit that is reached first is the policy-level, **limit address-count**. A maximum of 14 IP addresses (IPv4 and 1Pv6) are allowed on the port or interface where policy "sisf-01" is applied. No further binding entries are created after this. While the mac limit is configured with a lower absolute value (19), there are only 3 unique MAC address in the list of binding entries in the table - this limit is therefore not reached.

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# device-tracking policy sisf-01
Device(config-device-tracking)# limit address-count 14
Device(config-device-tracking)# end
Device# show device-tracking policy sisf-01
Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count 14
Policy sisf-01 is applied on the following targets:
Target               Type  Policy               Feature        Target range
Te1/0/4              PORT  sisf-01              Device-tracking vlan 200
```

After the stale lifetime of all the existing entries has expired and the entries have been removed from the binding table, new entries are added according to the reconfigured values:

```
Device# show device-tracking database  <<<<checking time left for stale-lifetime to expire
 for existing entries.
Binding Table has 20 entries, 20 dynamic (limit 30)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned

Network Layer Address                          Link Layer Address     Interface  vlan
prlvl      age        state     Time left
ARP 192.0.9.39                                 000c.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 316 s
ARP 192.0.9.38                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 279 s
ARP 192.0.9.37                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 308 s
ARP 192.0.9.36                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 274 s
ARP 192.0.9.35                                 000b.959d.6816         Te1/0/4    200
```

```
0064       13mn       STALE     try 0 279 s
ARP 192.0.9.34                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 261 s
ARP 192.0.9.33                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 258 s
ARP 192.0.9.32                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 263 s
ARP 192.0.9.31                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 266 s
ARP 192.0.9.30                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 273 s
ARP 192.0.9.29                                 000b.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 277 s
ARP 192.0.9.28                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 282 s
ARP 192.0.9.27                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 272 s
ARP 192.0.9.26                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 268 s
ARP 192.0.9.25                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 244 s
ARP 192.0.9.24                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 248 s
ARP 192.0.9.23                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 284 s
ARP 192.0.9.22                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 241 s
ARP 192.0.9.21                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 256 s
ARP 192.0.9.20                                 000a.959d.6816         Te1/0/4    200
0064       13mn       STALE     try 0 243 s

Device# show device-tracking database  <<<no output indicates no entries in the database


Device# show device-tracking database details

 Binding table configuration:
 ----------------------------
 max/box  : 30
 max/vlan : 25
 max/port : 20
 max/mac  : 19

 Binding table current counters:
 -------------------------------
 dynamic  : 14
 local    : 0
 total    : 14

 Binding table counters by state:
 --------------------------------
 REACHABLE  : 14
   total    : 14
<output truncated>

Device# show device-tracking database
Binding Table has 14 entries, 14 dynamic (limit 30)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned
```

```
Network Layer Address                          Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.68                                 0001.5e00.53af         Te1/0/4    200
0064       4s         REACHABLE  48 s
ARP 192.0.9.67                                 0001.5e00.53af         Te1/0/4    200
0064       4s         REACHABLE  48 s
ARP 192.0.9.66                                 0001.5e00.53af         Te1/0/4    200
0064       4s         REACHABLE  47 s
ARP 192.0.9.65                                 0001.5e00.53af         Te1/0/4    200
0064       4s         REACHABLE  48 s
ARP 192.0.9.64                                 0001.5e00.53af         Te1/0/4    200
0064       4s         REACHABLE  46 s
ARP 192.0.9.63                                 0000.5e00.53af         Te1/0/4    200
0064       7s         REACHABLE  44 s
ARP 192.0.9.62                                 0000.5e00.53af         Te1/0/4    200
0064       7s         REACHABLE  45 s
ARP 192.0.9.61                                 0000.5e00.53af         Te1/0/4    200
0064       7s         REACHABLE  43 s
ARP 192.0.9.60                                 0000.5e00.53af         Te1/0/4    200
0064       7s         REACHABLE  44 s
ARP 192.0.9.59                                 0000.5e00.53af         Te1/0/4    200
0064       7s         REACHABLE  44 s
ARP 192.0.9.58                                 0000.5e00.53af         Te1/0/4    200
0064       8s         REACHABLE  44 s
ARP 192.0.9.57                                 0000.5e00.53af         Te1/0/4    200
0064       8s         REACHABLE  44 s
ARP 192.0.9.56                                 0000.5e00.53af         Te1/0/4    200
0064       10s        REACHABLE  41 s
ARP 192.0.9.55                                 0000.5e00.53af         Te1/0/4    200
0064       10s        REACHABLE  40 s

Device# show device-tracking database mac
 MAC                    Interface  vlan       prlvl      state            Time left
 Policy           Input_index
 0001.5e00.53af         Te1/0/4    200        NO TRUST   MAC-REACHABLE    30 s
 sisf-01          12
 0000.5e00.53af         Te1/0/4    200        NO TRUST   MAC-REACHABLE    30 s
 sisf-01          12
```

### Example: Setting VLAN, Port, and MAC Limits to Default Values

The following example shows you how to reset one or more limits to their default values.

```
Device(config)# device-tracking binding max-entries 30 vlan-limit 25 port-limit 20 mac-limit
 19 <<<< all three limits configured.
Device(config)#exit
Device# show device-tracking database details

 Binding table configuration:
 ----------------------------
 max/box  : 30
 max/vlan : 25
 max/port : 20
 max/mac  : 19
<output truncated>

Device# configure terminal
Device(config)# device-tracking binding max-entries 30 vlan-limit 25 <<<< only VLAN limit
configured;  port-limit and mac-limit keywords leftout.
Device(config)# exit
Device# show device-tracking database details

 Binding table configuration:
 ----------------------------
```

```
 max/box  : 30
 max/vlan : 25
 max/port : no limit    <<<reset to default
 max/mac  : no limit    <<<reset to default
```

### Example: Global vs Policy-Level Limits Relating to MAC Addresses

The following example shows how precendence is determined for global and policy-level MAC limits. The global value specifies the maximum number of entries allowed per MAC address. The policy-level IPv4 per MAC and IPv6 per MAC limits, which may be present only in a programmatic policy, specify the number of IPv4 and IPv6 addresses allowed per MAC address.

In the first part of the example, the global value (10 entries allowed per MAC address) is higher than the policy-level setting (3 IPv4 addresses allowed for each MAC address). The `Binding table current counters`, in the output of the **show device-tracking database details** privileged EXEC command shows that and the limit that is reached first is the policy level limit.

**Note** No configuration is displayed for the policy-level setting, because you cannot *configure* the "IPv4 per mac" or the "IPv6 per mac" in any policy. In this example, the DT-PROGRAMMATIC policy is applied to target by configuring the **ip dhcp snooping vlan** *vlan* command in global configuration mode. The IPv4 per mac limit exists, because the programmatically created policy has a limit for this parameter.

```
Device# configure terminal
Device(config)# ip dhcp snooping vlan 200
Device(config)# end
Device# show device-tracking policy DT-PROGRAMMATIC
Policy DT-PROGRAMMATIC configuration:
  security-level glean (*)
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count for IPv4 per mac 3 (*)
  tracking enable
Policy DT-PROGRAMMATIC is applied on the following targets:
Target      Type    Policy               Feature            Target range
Te1/0/4     PORT    DT-PROGRAMMATIC      Device-tracking    vlan 200

  note:
  Binding entry Down timer: 24 hours (*)
  Binding entry Stale timer:   24 hours (*)

Device(config)# device-tracking binding max-entries 50 mac-limit 10
Device# show device-tracking database details
Binding table configuration:
 ----------------------------
 max/box  : 50
 max/vlan : no limit
 max/port : no limit
 max/mac  : 10

 Binding table current counters:
 -------------------------------
 dynamic  : 3
 local    : 0
 total    : 3
```

```
 Binding table counters by state:
 --------------------------------
 REACHABLE  : 2
   total    : 3

Device# show device-tracking database
Binding Table has 3 entries, 3 dynamic (limit 50)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


Network Layer Address                   Link Layer Address     Interface  vlan       prlvl
      age       state      Time left
ARP 192.0.9.8                            000a.959d.6816         Te1/0/4    200        0064
       4s        REACHABLE  25 s
ARP 192.0.9.7                            000a.959d.6816         Te1/0/4    200        0064
       4s        REACHABLE  27 s
ARP 192.0.9.6                            000a.959d.6816         Te1/0/4    200        0064
       55s       VERIFY     5s try 2
<<<<<<policy-level limit reached;  only up to 3 IPv4 addresses per MAC address are allowed.


Device# show device-tracking database mac
 MAC                   Interface  vlan       prlvl      state            Time left
 Policy           Input_index
 000a.959d.6816        Te1/0/4    200        NO TRUST   MAC-STALE        93585 s
 DT-PROGRAMMATIC          12
```

In the second part of the example, the global value (2 entries allowed per MAC address) is lower than the policy-level setting (3 IPv4 addresses allowed for each MAC address). The `Binding table current counters`, in the output of the **show device-tracking database details** privileged EXEC command shows that and the limit that is reached first is the policy level limit.

```
Device# show device-tracking policy DT-PROGRAMMATIC

Policy DT-PROGRAMMATIC configuration:
  security-level glean (*)
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count for IPv4 per mac 3 (*)
  tracking enable
Policy DT-PROGRAMMATIC is applied on the following targets:
Target      Type    Policy               Feature            Target range
Te1/0/4     PORT    DT-PROGRAMMATIC      Device-tracking    vlan 200

  note:
  Binding entry Down timer: 24 hours (*)
  Binding entry Stale timer:   24 hours (*)

Device(config)# device-tracking binding max-entries 50 mac-limit 2
Device# show device-tracking database details
Binding table configuration:
 ---------------------------
```

```
 max/box  : 50
 max/vlan : no limit
 max/port : no limit
 max/mac  : 2

 Binding table current counters:
 -------------------------------
 dynamic  : 2
 local    : 0
 total    : 2

 Binding table counters by state:
 ---------------------------------
 REACHABLE  : 2
   total    : 2

Device# show device-tracking database
Binding Table has 3 entries, 3 dynamic (limit 50)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


Network Layer Address                   Link Layer Address     Interface  vlan       prlvl
      age       state      Time left
ARP 192.0.9.3                            000a.959d.6816         Te1/0/4    200        0064
       5s        REACHABLE   27 s
ARP 192.0.9.4                            000a.959d.6816         Te1/0/4    200        0064
       6s        REACHABLE   20 s

<<<<<<global limit reached;  only up to 2 binding entries per MAC address is allowed.

Device# show device-tracking database mac
 MAC                   Interface  vlan       prlvl      state            Time left
 Policy           Input_index
 000a.959d.6816         Te1/0/4    200        NO TRUST   MAC-STALE        93585 s
 DT-PROGRAMMATIC       12
```

# device-tracking logging

To log snooping security events like packet drops, unresolved packets, and suspected MAC or IP theft, configure the **device-tracking logging** command in global configuration mode. To disable logging, enter the **no** form of the command.

**device-tracking logging** [ **packet drop** | **resolution-veto** | **theft** ]

**no device-tracking logging** [ **packet drop** | **resolution-veto** | **theft** ]

## Syntax Description

| | |
|---|---|
| **packet drop** | Logs packet drop events. |
| **resolution-veto** | Logs unresolved packet events. |
| **theft** | Logs IP and MAC theft events. |

## Command Default

Events are not logged.

## Command Modes

Global configuration [Device(config)# ]

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Logs generated for snooping security events have a severity level of 4 (meaning, warnings). For example:

```
%SISF-4-PAK_DROP: Message dropped A=FE80::20D:FF:FE0E:F G=- V=10 I=Tu0 P=NDP::RA Reason=Packet not authorized on port
```

You can view snooping security logs by entering the **show logging | include SISF-4** command in privileged EXEC mode.

For information about the snooping events for which logs are generated, see the system message guide for the corresponding release: System Message Guides. Search for `SISF-4`.

### Packet Drop Events

When you configure the **packet drop** keyword, a log is generated everytime a packet is dropped. The log also includes the reason for the packet drop. The reasons include and are not limited to the following:

- `Packet not authorized on port`: This means that a security feature dropped the packet because a packet of this kind is not expected on the port, based on the configuration. Examples of such security features and the situations in which a packet is dropped, include and are not limited to the following: The Router Advertisement Guard feature may decide to drop IPv6 Router Advertisement packets if they are received on ports that are not configured as router-facing ports. The DHCP Guard feature may drop packets from DHCP server (DHCP OFFER or DHCP REPLY) if they are received on a port which is not configured as server-facing port.
- `Packet accepted but not forwarded`: This means that the packet is not forwarded, but it is still considered valid to glean binding information from. This is usually seen when packets from a host are seen by SISF during the validation phase (while the binding is in a transitional state).

- `Malformed Packet dropped in Guard mode`: This means that the incoming packet is malformed and cannot be parsed properly.
- `Packet is throttled`: This means the packet was dropped because it exceeds the throttling limit for packets within a time interval. The system allows a maximum of 50 packets in 5 seconds.
- `Silent drop`: This happens to packets that are generated either by device-tracking instances to communicate among the different instances across multiple switches, or as a response to an action trigged by device-tracking. For instance, a response on the probe that was initiated by the device-tracking, to determine the reachability status of the host reachability.
- `Martian packet`: This means that the incoming packet was dropped because it has Martian source IP address, such as, a multicast, loopback, or unspecified address.
- `Martian mac`: This means that the incoming packet was dropped because it has a Martian MAC or Link-Layer source address.
- `Address limit per box reached`: This means that the incoming packet was dropped, because the limit configured with the **device-tracking binding max-entries** *no_of_entries* global configuration command, was reached. Enter the **show device-tracking database details** privileged EXEC command to display current limits.
- `Address limit per vlan reached`: This means that the incoming packet was dropped, because the limit configured with the **device-tracking binding max-entries** *no_of_entries* **vlan-limit** *no_of_entries* global configuration command, was reached. Enter the **show device-tracking database details** privileged EXEC command to display current limits.
- `Address limit per port reached`: This means that the incoming packet was dropped, because the limit configured with the **device-tracking binding max-entries** *no_of_entries* **port-limit** *no_of_entries* global configuration command, was reached. Enter the **show device-tracking database details** privileged EXEC command to display current limits.
- `Address limit per policy reached` : This means that the incoming packet was dropped, because the limit configured with the **limit address-count** *ip-per-port* keyword in the device-tracking configuration mode was reached. This is configured at a policy level. Enter the **show device-tracking policy***policy-name* privileged EXEC command to display current limits.
- `Address limit per mac reached`: This means that the incoming packet was dropped, because the limit configured with the **device-tracking binding max-entries** *no_of_entries* **mac-limit** *no_of_entries* global configuration command, was reached. Enter the **show device-tracking database details** privileged EXEC command to display current limits.
- `Address Family limit per mac reached`: This means that the incoming packet was dropped, because the IPv4 per MAC or IPv6 per MAC limit specified in a programmatic policy was reached. You cannot configure this policy parameter; a programmatically created policy may have either an IPv4 per MAC limit, or an IPv6 per MAC limit, or both, or neither. Enter the **show device-tracking policy***policy-name* privileged EXEC command to display the limit if it exists.

### Resolution Veto Events

When you configure the **resolution-veto** keyword, a log is generated for every unresolved packet. This logging option meant to be used only if the IPv6 Destination Guard feature is also enabled.

The IPv6 Destination Guard feature ensures that the device performs address resolution only for those addresses that are known to be active on the link. All destinations that are active on the link are entered in the binding

table. When a destination is not found in the binding table, address resolution is prevented. By configuring **resolution-veto** logging you can keep track of such unresolved packets.

If the **resolution-veto** keyword is configured and the the IPv6 Destination Guard feature is not, logs are not generated.

**Theft Events**

When you configure the **theft** keyword, a log is generated when SISF detects an IP theft, or a MAC theft or both.

In the log, verified binding information (IP, MAC address, interface or VLAN) is preceded by the term "Known" . A suspicious IP address and MAC address is preceded by the term "New" or "Cand". Interface and VLAN information is also provided along with the suspiscious IP or MAC address - this helps you identify where the suspiscious traffic was seen.

For example, see the following MAC theft log:

```
%SISF-4-MAC_THEFT: MAC Theft Cand IP=2001::12B VLAN=70 MAC=9cfc.e85e.139d Cand I/F=Gi1/0/4
 Known IP=71.0.0.96 Known I/F=Ac0
```

These snippets of the log show the IP address of the suspiscious host and the interface on which it was seen: `Cand IP=2001::12B`, `VLAN=70`, `Cand I/F=Gi1/0/4`.

This snippet of the log shows the *known* MAC address, which the suspiscious host is using: `MAC=9cfc.e85e.139d`.

These snippets of the log show the IP address and interface of the existing, verified entry: `Known IP=71.0.0.96` and `Known I/F=Ac0`.

### Examples

- Example: Packet Drop Logs, on page 91
- Example: Theft Logs, on page 91

### Example: Packet Drop Logs

The following are examples of logs generated for packet drop events:

```
%SISF-4-PAK_DROP: Message dropped A=FE80::20D:FF:FE0E:F G=- V=10 I=Tu0 P=NDP::RA Reason=Packet
 not authorized on port

%SISF-4-PAK_DROP: Message dropped A=20.0.0.1 M=dead.beef.0001 V=20 I=Gi1/0/23 P=ARP
Reason=Packet accepted but not forwarded
```

### Example: Theft Logs

The following are examples of logs generated for IP and MAC theft events:

```
%SISF-4-MAC_AND_IP_THEFT: MAC_AND_IP Theft A=FE80::EE1D:8BFF:FE9B:102 V=102 I=Vl102
M=ec1d.8b9b.0102 New=Tu0

%SISF-4-MAC_THEFT: MAC Theft IP=192.2.1.2 VLAN=102 MAC=cafe.cafe.cafe I/F=Gi1/0/3 New I/F
over fabric

%SISF-4-IP_THEFT: IP Theft IP=FE80::9873:1D5E:E6E9:1F7E VLAN=20 MAC=2079.18d5.13ad IF=Ac0
New I/F over fabric
```

```
%SISF-4-IP_THEFT: IP Theft IP=10.0.187.5 VLAN=10 Cand-MAC=0069.0000.0001 Cand-I/F=Gi1/0/23
 Known MAC over-fabric Known I/F over-fabric

%SISF-4-MAC_THEFT: MAC Theft Cand IP=2001::12B VLAN=70 MAC=9cfc.e85e.139d Cand I/F=Gi1/0/4
 Known IP=71.0.0.96 Known I/F=Ac0
```

# device-tracking policy

To create a custom device-tracking policy, and to enter the device-tracking configuration mode to configure the various parameter of the policy, enter the **device-tracking policy** command in global configuration mode. To delete a device tracking policy, use the **no** form of this command.

**device-tracking policy** *policy-name*
**no device-tracking policy** *policy-name*

## Syntax Description

| | |
|---|---|
| *policy-name* | Creates a device-tracking policy with the specified name - if it doesn't already exist. You can also specify the name of a programmatically created policy.<br><br>After you configure a policy name, the device enters the device-tracking configuration mode, where you can configure policy parameters. Enter a question mark (?) at the system prompt to see the list of policy parameters that can be configured. |

## Command Default

SISF-based device tracking is disabled.

## Command Modes

Global configuration [Device(config)# ]

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Everest 16.6.1 | Option to change certain parameters of programmatic policy DT_PROGRAMMATIC was introduced. |
| Cisco IOS XE Fuji 16.9.1 | The option to change the parameters of *any* programmatic policy was deprecated. |
| Cisco IOS XE 17.13.1 | The **udp** keyword was deprecated. There is no replacement.<br><br>The **udp** keyword was one of the options available with the **protocol** keyword. |

## Usage Guidelines

When you enter the **device-tracking policy** *policy-name* command in global configuration mode, the system creates a custom policy with the specified name (if it does not already exist) and enters the device-tracking configuration mode. In this mode, you can configure policy parameters.

After you create a policy and configure its parameters, you must attach it to an interface or VLAN. Only then does the activity of extracting binding information (IP and MAC address) from packets that enter the network and the creation of binding entries, actually begin. For more information about attaching a policy, see device-tracking (interface config), on page 63device-tracking (VLAN config), on page 66.

To display detailed information about all the policies available on the device and the targets they are attached to, enter the **show device-tracking policies detail** command in privileged EXEC mode.

### Configuring Policy Parameters

You can configure the parameters of a policy only if it is a custom policy. You cannot change the parameters of a programmatic policy. You also cannot change the parameters of the `default` policy.

To display the list of parameters for a policy, enter a question mark (?) at the system prompt in device-tracking configuration mode:

```
Device(config)# device-tracking policy sisf-01
Device(config-device-tracking)# ?
device-tracking policy configuration mode:
  data-glean            binding recovery by data traffic source address
                        gleaning
  default               Set a command to its defaults
  destination-glean     binding recovery by data traffic destination address
                        gleaning
  device-role           Sets the role of the device attached to the port
  distribution-switch   Distribution switch to sync with
  exit                  Exit from device-tracking policy configuration mode
  limit                 Specifies a limit
  medium-type-wireless  Force medium type to wireless
  no                    Negate a command or set its defaults
  prefix-glean          Glean prefixes in RA and DHCP-PD traffic
  protocol              Sets the protocol to glean (default all)
  security-level        setup security level
  tracking              Override default tracking behavior
  trusted-port          setup trusted port
  vpc                   setup vpc port
```

| Keyword | Description |
|---|---|
| **data-glean** | Enables learning of addresses from a data packet snooped from a source inside the network and populates the binding table with the data traffic source address. Enter one of these options:<br>• **log-only**: Generates a syslog message upon data packet notification.<br>• **recovery**: Uses a protocol to enable binding table recovery. Enter **NDP** or **DHCP**. |
| **default** | Sets the policy paramter to its default value. You can set these policy attributes to their default values:<br>• **data-glean**: Source address is not learnt or gleaned.<br>• **destination-glean**: Destination address is not learnt or gleaned<br>• **device-role**: Node.<br>• **distribution-switch**: Not supported.<br>• **limit**: An address count limit is not set.<br>• **medium-type-wireless**: <tbd><br>• **prefix-glean**: Prefixes are not learnt.<br>• **protocol**: Addresses of all protocols (ARP, DHCP4, DHCP6, and NDP) are gleaned.<br>• **security-level**: Guard.<br>• **tracking**: Polling is disabled.<br>• **trusted-port**: Disabled, that is, the guard function is enabled on the configured target)<br>• **vpc**: Not supported. |

| Keyword | Description |
|---|---|
| **destination-glean** | Enables population of the binding table by gleaning the destination address of data traffic. Enter one of these options:<br>• **log-only**: Generates a syslog message upon data packet notification.<br>• **recovery**: Uses a protocol to enable binding table recovery. Enter **NDP** or **DHCP**. |
| **device-role** | Indicates the type of device that is facing the port and this can be one of the following:<br>• **node**: Allows creation of binding entries for a port.<br>• **switch**: Stops the creation of binding entries for a port. This option is suited to multi-switch set-ups, where the possibility of large device tracking tables is very high. Here, a port facing a device (an uplink trunk port) can be configured to stop creating binding entries, and the traffic arriving at such a port can be trusted, because the switch on the other side of the trunk port will have device tracking enabled and that will have checked the validity of the binding entry.<br>This option is commonly used along with the **trusted-port** keyword. Configuring both the **device-role** and **trusted-port** options on an uplink trunk port helps build an efficient and scalable "secure zone". Both parameters must be configured to achieve an efficient distribution of the creation of binding table entries (thus keeping the binding tables smaller). |
| **distribution-switch** | Although visible on the CLI, this keyword is not supported. Any configuration does not take effect. |
| **exit** | Exits the device-tracking configuration mode and returns to global configuration mode. |
| **limit address-count** | Configures the maximum number of number of IPv4 and IPv6 addresses to be allowed per port. The purpose of this limit is to ensure that binding entries are restricted to only known and expected hosts.<br>*ip-per-port*: Enter the maximum number of IP addresses you want to allow on a port. This limit applies to IPv4 and IPv6 addresses as a whole. When the limit is reached, no further IP addresses can be added to the binding table, and traffic from new hosts are dropped.<br>Enter a value between 1 and 32000. |
| **medium-type-wireless** | Although visible on the CLI, this keyword is not supported. Any configuration does not take effect. |

| Keyword | Description |
|---|---|
| **no** | Negates the command, that is, reverts a policy parameter to its default value.<br>For information about what the default value is, see the **default** keyword.<br>• **data-glean**<br>• **destination-glean**<br>• **device-role**<br>• **distribution-switch**: Not supported.<br>• **limit address-count**<br>• **medium-type-wireless**<br>• **prefix-glean**<br>• **protocol**<br>• **security-level**<br>• **tracking**<br>• **trusted-port**<br>• **vpc**: Not supported. |
| **prefix-glean only** | Enables learning of prefixes from either IPv6 Router Advertisements or from DHCP-PD. You have the following option:<br>(Optional) **only**: Gleans only prefixes and not host addresses. |
| **protocol** | Gleans addresses of specified protocols. By default, all are gleaned.<br>• **arp** [**prefix-list** *name*]: Gleans addresses in ARP packets. Optionally, enter the name of prefix-list that is to be matched.<br>• **dhcp4** [**prefix-list** *name*]: Gleans addresses in DHCPv4 packets. Optionally, enter the name of prefix-list that is to be matched.<br>• **dhcp6** [**prefix-list** *name*]: Gleans addresses in DHCPv6 packets. Optionally, enter the name of prefix-list that is to be matched.<br>• **ndp** [**prefix-list** *name*]: Gleans addresses in NDP packets. Optionally, enter the name of prefix-list that is to be matched. |

| Keyword | Description |
|---|---|
| **security-level** | Specifies the level of security that is enforced. When a packet enters the network, SISF extracts the IP and MAC address (the source of the packet) and subsequent action, is dictated by the security level configured in the policy.<br>Enter one of these options:<br>• **glean**: Extracts the IP and MAC address and enters them into the binding table, without any verification. Use this option if you want to only *learn* about the host and not rely on SISF for authentication of the binding entry.<br>• **guard**: Extracts the IP and MAC address and checks this information against the binding table. The outcome of the verification determines if a binding entry is added, or updated, or if the packet is dropped and the client is rejected<br>This is the default value for the security-level parameter.<br>• **inspect**: Although this keyword is available on the CLI, we recommend not using it. The **glean** and **guard** options described above address most use cases and network requirements. |

| Keyword | Description |
|---|---|
| **tracking** | Determines if an entry is polled after the reachable lifetime expires. Polling is a periodic and conditional checking of the host to see the state it is in, whether it is still connected, and whether it is communicating. For more information about polling, see the *Usage Guidelines* below.<br><br>By default, polling is not enabled.<br><br>Enter one of these options:<br>• **disable** : Turns off polling action.<br>[**stale-lifetime** {*seconds* \| **infinite**} ]: Optionally you can also configure a stale-lifetime. If you do, configure one of the following for the stale-lifetime timer:<br>• *seconds*: Configure a value for the stale-lifetime timer. Enter a value between 1 and 86400 seconds. The default value is 86400 seconds (24 hours).<br>• **infinite**: Disables the timer for the STALE state. This means that a timer is not started when an entry enters the STALE state and the entry remains in the STALE state, indefinitely.<br>• **enable**: Turns on polling action.<br>[**reachable-lifetime** [*seconds* \| **infinite**] ]: Optionally you can also configure a reachable-lifetime. If you do, configure one of the following for the reachable-lifetime timer:<br>• *seconds*: Configure a value for the reachable-lifetime timer. Enter a value between 1 and 86400 seconds. The default value is 300 seconds (5 minutes).<br>• **infinite**: Disables the timer for the REACHABLE state. This means that a timer is not started when an entry enters the REACHABLE state and the entry remains in the REACHABLE state, indefinitely. |
| **trusted-port** | This option disables the guard function on configured targets. Bindings learned through a trusted-port have preference over bindings learned through any other port. A trusted port is also given preference in case of a collision while making an entry in the table.<br><br>This option is commonly used along with the **device-role** keyword. Configuring both the **device-role** and **trusted-port** options on an uplink trunk port helps achieve an efficient distribution of the creation of binding table entries (thus keeping the binding tables smaller). |
| **vpc** | Although visible on the CLI, this option is not supported. Any configuration does not take effect. |

**Global versus Poicy-Level Settings**

You configure policy parameters in the device-tracking configuration mode and what you configure for a policy applies only to that policy. Some of the policy parameters have counterparts in the global configuration mode. For detailed information about the parameters that have global-level counterparts and to know which value takes precedence (whether the globally configured or the policy-level value), see: device-tracking binding, on page 69.

**Polling a Host**

If you configure the **tracking** policy parameter, the switch sends a polling request after the reachable lifetime expires. The switch polls the host up to 3 times at fixed, system-determined intervals. You can also specify an interval by using the **device-tracking tracking retry-interval** *seconds* command in global configuration mode. The polling request is in the form of an Address Resolution Protocol (ARP) probe or a Neighbor Solicitation message. During this time the state of the entry changes to VERIFY.

If a polling response is received (thus confirming reachability of the host), the state of the entry changes back to REACHABLE. If the switch does not receive a polling response after 3 attempts, the entry changes to the STALE state.

**Note** Using the **tracking** policy parameter, you can enable or disable polling at a policy-level regardless of whether the polling is enabled or disabled at the global configuration level (the **device-tracking tracking** command in global configuration mode. See Example: Disabling Polling at a Policy-Level, on page 100and device-tracking tracking, on page 106.

**Changing the Limit Address-Count**

If you configure a limit using the **limit address-count** policy parameter and then change it - the new limit is applicable only to entries learned after the change. Further, regardless of whether the new limit is higher or lower than the previous limit, existing entries are not affected and are allowed to go through their binding entry lifecycle.

If the binding table is full (in accordance with the previous limit), any new entries are not added until the existing entries complete their lifecycle. SISF attempts to create space for new entries by identifying and removing only *inactive* entries . But if the entries are active, they are not removed and are allowed to go through their binding entry lifecycle.

If you want to make the new lower limit take effect immediately, you can use either one of these options:

- Enter the **clear device-tracking database** command in privileged EXEC mode and specify an interface or VLAN. This removes all existing entries from the database of only the specified target. New entries are then learned and added as per the current limit address-count settings. See Example: Changing the Address Count Limit, on page 100.
- Remove and reattach the policy on the required target. Enter the **no device-tracking policy***policy-name* command in interface or VLAN configuration mode to remove the policy. Removing the policy from an interface or VLAN removes the bindings that are attached to the target. Enter the **device-tracking policy***policy-name* command in interface or VLAN configuration mode to reattach it. Reattaching the policy causes learning of all the binding entries according to the new limit.

## Examples

- Example: Disabling Polling at a Policy-Level, on page 100
- Example: Changing the Address Count Limit, on page 100

### Example: Disabling Polling at a Policy-Level

The following example shows how you can disable polling at the policy-level even if polling is enabled at the global level. Here, polling is disabled for all interfaces and VLANs were policy `sisf-01` is applied.

```
Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# device-tracking tracking
Device(config)# exit
Device# show running-config | include device-tracking device-tracking tracking
device-tracking policy sisf-01
 device-tracking attach-policy sisf-01
 device-tracking attach-policy sisf-01 vlan 200
device-tracking binding reachable-lifetime 700 stale-lifetime 1000 down-lifetime 200
device-tracking binding logging

Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# device-tracking policy sisf-01
Device(config-device-tracking)# tracking disable
Device(config-device-tracking)# end
Device# show device-tracking policy sisf-01
Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count 5
  tracking disable
Policy sisf-01 is applied on the following targets:
Target               Type  Policy               Feature        Target range
Te1/0/4              PORT  sisf-01              Device-tracking vlan 200
vlan 200             VLAN  sisf-01              Device-tracking vlan all
```

### Example: Changing the Address Count Limit

The following example shows you how to make a change in the **limit address-count** policy parameter setting take effect immediately. In this example, the clear command is used to remove all entries from the binding table for the changed settings to take effect immediately.

```
Device# show device-tracking policy sisf-01
Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count 25
Policy sisf-01 is applied on the following targets:
Target               Type  Policy               Feature        Target range
Te1/0/4              PORT  sisf-01              Device-tracking vlan 200
vlan 200             VLAN  sisf-01              Device-tracking vlan all

Device# show running-config | include device-tracking
```

```
device-tracking policy sisf-01
 device-tracking attach-policy sisf-01
 device-tracking attach-policy sisf-01 vlan 200
device-tracking binding reachable-lifetime 700 stale-lifetime 1000 down-lifetime 200
device-tracking binding logging


*Dec 13 15:08:50.723: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.25 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.723: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.26 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.724: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.27 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.724: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.28 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.724: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.29 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.724: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.30 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.725: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.31 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.725: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.32 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.725: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.33 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.725: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.34 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.726: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.35 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.726: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.36 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.726: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.37 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.726: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.38 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.727: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.39 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.727: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.40 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.727: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.41 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.727: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.42 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.728: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.43 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.728: %SISF-6-ENTRY_MAX_ORANGE: Reaching 80% of max adr allowed per policy
 (25) V=200 I=Te1/0/4 M=001d.4411.3ab7
*Dec 13 15:08:50.728: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.44 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.728: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.45 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.728: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.46 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.729: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.47 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.729: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.48 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:08:50.729: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.49 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF

Device# show device-tracking database Binding Table has 25 entries, 25 dynamic (limit 200000)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
```

```
0001:MAC and LLA match     0002:Orig trunk           0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access  0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated   0100:Statically assigned

    Network Layer Address                  Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.49                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  699 s
ARP 192.0.9.48                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  691 s
ARP 192.0.9.47                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  687 s
ARP 192.0.9.46                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  714 s
ARP 192.0.9.45                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  692 s
ARP 192.0.9.44                             001d.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  702 s
ARP 192.0.9.43                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  680 s
ARP 192.0.9.42                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  708 s
ARP 192.0.9.41                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  683 s
ARP 192.0.9.40                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  708 s
ARP 192.0.9.39                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  710 s
ARP 192.0.9.38                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  697 s
ARP 192.0.9.37                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  707 s
ARP 192.0.9.36                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  695 s
ARP 192.0.9.35                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  708 s
ARP 192.0.9.34                             001c.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  706 s
ARP 192.0.9.33                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  683 s
ARP 192.0.9.32                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  697 s
ARP 192.0.9.31                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  683 s
ARP 192.0.9.30                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  678 s
ARP 192.0.9.29                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  696 s
ARP 192.0.9.28                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  704 s
ARP 192.0.9.27                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  713 s
ARP 192.0.9.26                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  695 s
ARP 192.0.9.25                             001b.4411.3ab7         Te1/0/4    200
00FF       22s        REACHABLE  686 s
```

The address count limit is changed from 25 to a lower limit of 5. But because the existing entries have not completed their binding entry lifecycle, they are not deleted from the binding table. In order to make the new address count limit of 5 take effect immediately, the **clear device-tracking database** command is used to delete all existing entries. New entries are then learned and added as per the current limit address-count settings.

```
Device# configure terminal
Device(config)# device-tracking policy sisf-01
Device(config-device-tracking)# limit address-count 5
Device(config-device-tracking)# end
Device# show device-tracking policy sisf-01
Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  limit address-count 5
Policy sisf-01 is applied on the following targets:
Target              Type  Policy              Feature        Target range
Te1/0/4             PORT  sisf-01             Device-tracking vlan 200
vlan 200            VLAN  sisf-01             Device-tracking vlan all

Device# show device-tracking database
Binding Table has 25 entries, 25 dynamic (limit 200000)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


    Network Layer Address                    Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.49                               001d.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  654 s
ARP 192.0.9.48                               001d.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  646 s
ARP 192.0.9.47                               001d.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  642 s
ARP 192.0.9.46                               001d.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  669 s
ARP 192.0.9.45                               001d.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  647 s
ARP 192.0.9.44                               001d.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  657 s
ARP 192.0.9.43                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  635 s
ARP 192.0.9.42                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  663 s
ARP 192.0.9.41                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  638 s
ARP 192.0.9.40                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  663 s
ARP 192.0.9.39                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  665 s
ARP 192.0.9.38                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  652 s
ARP 192.0.9.37                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  662 s
ARP 192.0.9.36                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  650 s
ARP 192.0.9.35                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  663 s
ARP 192.0.9.34                               001c.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  661 s
```

```
ARP 192.0.9.33                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  637 s
ARP 192.0.9.32                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  652 s
ARP 192.0.9.31                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  638 s
ARP 192.0.9.30                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  633 s
ARP 192.0.9.29                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  651 s
ARP 192.0.9.28                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  658 s
ARP 192.0.9.27                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  668 s
ARP 192.0.9.26                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  650 s
ARP 192.0.9.25                            001b.4411.3ab7         Te1/0/4    200
00FF       67s        REACHABLE  641 s

Device# clear device-tracking database

*Dec 13 15:10:22.837: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.49 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.838: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.48 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.838: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.47 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.838: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.46 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.839: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.45 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.839: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.44 VLAN=200
MAC=001d.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.839: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.43 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.839: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.42 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.840: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.41 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.840: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.40 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.840: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.39 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.841: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.38 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.841: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.37 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.841: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.36 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.842: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.35 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.842: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.34 VLAN=200
MAC=001c.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.842: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.33 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.842: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.32 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.843: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.31 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.843: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.30 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.843: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.29 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.844: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.28 VLAN=200
```

```
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.844: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.27 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.844: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.26 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:10:22.844: %SISF-6-ENTRY_DELETED: Entry deleted IP=192.0.9.25 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF

Device# show device-tracking database
<no output; binding table cleared>

*Dec 13 15:11:38.346: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.25 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:11:38.346: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.26 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:11:38.347: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.27 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:11:38.347: %SISF-6-ENTRY_MAX_ORANGE: Reaching 80% of max adr allowed per policy
 (5) V=200 I=Te1/0/4 M=001b.4411.3ab7
*Dec 13 15:11:38.347: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.28 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF
*Dec 13 15:11:38.347: %SISF-6-ENTRY_CREATED: Entry created IP=192.0.9.29 VLAN=200
MAC=001b.4411.3ab7 I/F=Te1/0/4 Preflevel=00FF

Device# show device-tracking database
Binding Table has 5 entries, 5 dynamic (limit 200000)
Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


    Network Layer Address                    Link Layer Address     Interface  vlan
prlvl      age        state      Time left
ARP 192.0.9.29                               001b.4411.3ab7         Te1/0/4    200
00FF       15s        REACHABLE  716 s
ARP 192.0.9.28                               001b.4411.3ab7         Te1/0/4    200
00FF       15s        REACHABLE  702 s
ARP 192.0.9.27                               001b.4411.3ab7         Te1/0/4    200
00FF       15s        REACHABLE  705 s
ARP 192.0.9.26                               001b.4411.3ab7         Te1/0/4    200
00FF       15s        REACHABLE  716 s
ARP 192.0.9.25                               001b.4411.3ab7         Te1/0/4    200
00FF       15s        REACHABLE  718 s
```

# device-tracking tracking

To enable polling for IPv4 and IPv6 and configure the polling parameters, configure the **device-tracking tracking** command in global configuration mode. To disable polling, enter the **no** form of the command.

**Note** This command does not enable the SISF-based device-tracking feature. It enables configuration of polling parameters on a device where the device-tracking feature is enabled.

**device-tracking tracking** [ **auto-source** [ **fallback** *ipv4_and_fallback_source_mask ip_prefix_mask* [ **override** ] | **retry-interval** *seconds* ]

**no device-tracking tracking** [ **auto-source** | **retry-interval** ]

## Syntax Description

| | |
|---|---|
| **auto-source** | Causes the source address of an Address Resolution Protocol (ARP) probe to be sourced in the following order of preference:<br>• The first preference is to set the source address to the VLAN SVI, if an SVI is configured.<br>• The second preference is to locate an IP-MAC binding entry in device-tracking table, from same subnet and use that as the source address.<br>• The third and last preference is to use 0.0.0.0 as the source address. |
| **fallback** *ipv4_and_fallback_source_maskip_prefix_mask* | Causes the source address of an ARP probe to be sourced in the following order of preference:<br>• The first preference is to set the source address to the VLAN SVI, if an SVI is configured.<br>• The second preference is to locate an IP-MAC binding entry in device-tracking table, from same subnet and use that as the source address.<br>• The third and last preference is to compute the source address from the client's IPv4 address and the mask provided.<br>The source MAC address is taken from the MAC address of the switchport facing the client.<br>If you configure the **fallback** keyword, you must also specify an IP address and mask. |

| | |
|---|---|
| **override** | Causes the source address of an ARP probe to be sourced in the following order of preference:<br>• The first preference is to set the source address to the VLAN SVI, if this is configured.<br>• The second and last preference is to use 0.0.0.0 as the source address.<br>**Note**<br>This keyword configures SISF to *not* select the source address from the binding table. We do not recommend using this option if an SVI is not configured. |
| **retry-interval** *seconds* | Configures a multiplicative factor or "base value", for the backoff algorithm. The backoff algorithm determines the wait time between the 3 polling attempts that occur after reachable lifetime expiry.<br>Enter a value between 1 and 3600 seconds. The default value is one.<br>When polling, there is an increasing wait time between the 3 polling attempts or retries. The backoff algorithm determines this wait time. The value you configure for the retry interval is multiplied by the backoff algorithm's wait time.<br>For example, if the backoff algorithm determines a wait time of 2, 4, and 6 seconds between the 3 attempts respectively, and you configure a retry interval of 2 seconds, the actual interval you will observe is as follows: 2*2 seconds of wait time before the first polling attempt, 2*4 seconds for the second polling attempt and 2*6 for the third polling attempt.<br>If polling is enabled, but a retry interval is not configured, the switch polls the host up to 3 times at system-determined intervals.<br>This configuration applies to ARP probes and Neighbor Solicitation messages. |

**Command Default**

Polling is disabled by default.

**Command Modes**

Global configuration [Device(config)# ]

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Polling is a periodic and conditional checking of the host to see the state it is in, whether it is still connected, and whether it is communicating. Polling enables you to assess the continued presence of a tracked device.

Polling occurs at these junctures: 3 times after the reachable lifetime timer expires, and a final attempt at stale lifetime expiry.

- In an IPv4 network, polling is in the form of an ARP probe. Here, the switch sends unicast ARP probes to the connected host, to determine the host's reachability status. When sending ARP probes, the system constructs packets according to RFC 5227 specifications.
- In an IPv6 network, polling is in the form of a Neighbor Solicitation message. Here, the switch verifies reachability of a connected host by using the unicast address of the connected host as the destination address.

Configure the **device-tracking tracking** command in global configuration mode, to enable polling for IPv4 and IPv6.

Also configure the **retry-interval** *seconds* to configure the polling interval after reachable lifetime timer expiry.

**Note** The **auto-source**, **fallback** *ipv4_and_fallback_source_maskip_prefix_mask* , and **override** keywords apply only to ARP probes and not Neighbor Solicitation messages.

The value you configure for **retry-interval** *seconds* keywords applies to both IPv4 and IPv6.

Enter the **show running-config | include device-tracking** display current polling settings. For example:

```
Device# show running-config | include device-tracking
device-tracking tracking retry-interval 2
device-tracking policy sisf-01
 device-tracking attach-policy sisf-01 vlan 200
device-tracking binding reachable-lifetime 50 stale-lifetime 150 down-lifetime 30
device-tracking binding logging
```

Enter the **show device-tracking database** command in privileged EXEC mode, to display the duration of the various lifetimes of an entry. While polling, the system changes the state of the entry to VERIFY. Check the `Time left` column in the output to observe the duration.

When you track the reachable and stale lifetime of an entry with the **show device-tracking database** command, and polling is enabled, you may notice that the STALE lifetime is sometimes shorter than what you have configured. This is because the time required for polling is *subtracted* from the stale lifetime.

### Global versus Policy-Level Settings for Polling

After you configure **device-tracking tracking** command in global configuration mode, you still have the flexibility to turn polling on or off, for individual interfaces and VLANs. For this you must enable or disable polling in the policy. Note how the global and policy-level settings interact:

| Global Setting | Policy-Level Setting | Result |
|---|---|---|
| Polling is enabled at the global level.<br>`Device(config)# device-tracking tracking` | Polling is enabled on an interface or VLAN.<br>`Device(config-device-tracking)# tracking enable` | Polling is effective on the interface or VLAN. |
| | Polling is disabled on an interface or VLAN.<br>`Device(config-device-tracking)# tracking disable` | Polling is not effective on the interface or VLAN. |
| | Default polling is configured on the interface or VLAN.<br>`Device(config-device-tracking)# default tracking` | Because polling is enabled at the *global* config level, polling is effective on the interface or VLAN. |
| | The **no** form of the command is configured on the interface or VLAN.<br>`Device(config-device-tracking)# no tracking` | The **no** form of the command sets the command to its default. But because polling is enabled at the *global* config level, polling is effective on the interface or VLAN. |
| Polling is disabled at the global level.<br>`Device(config)# no device-tracking tracking` | Polling is enabled on an interface or VLAN.<br>`Device(config-device-tracking)# tracking enable` | Polling is effective on the interface or VLAN. |
| | Polling is disabled on an interface or VLAN.<br>`Device(config-device-tracking)# tracking disable` | Polling is not effective on the interface or VLAN. |
| | Default polling is configured on the interface or VLAN.<br>`Device(config-device-tracking)# default tracking` | Polling is not effective on the interface or VLAN. |
| | The **no** form of the command is configured on the interface or VLAN.<br>`Device(config-device-tracking)# no tracking` | Polling is not effective on the interface or VLAN. |

# device-tracking upgrade-cli

To convert legacy IP Device Tracking (IPDT) and IPv6 Snooping commands to SISF commands, configure the **device-tracking upgrade-cli** command in global configuration mode. To revert to legacy commands, enter the **no** form of the command.

**device-tracking upgrade-cli** [ **force** | **revert** ]

**no device-tracking upgrade-cli** [ **force** | **revert** ]

## Syntax Description

| | |
|---|---|
| **force** | Skips the confirmation step and converts legacy IPDT and IPv6 Snooping commands to SISF commands. |
| **revert** | Reverts to legacy IPDT and IPv6 Snooping commands. |

## Command Default

Legacy IPDT and IPv6 Snooping commands remain as-is.

## Command Modes

Global configuration [Device(config)# ]

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

Based on the legacy configuration that exists on your device, the **device-tracking upgrade-cli** command upgrades your CLI differently. Consider the following configuration scenarios and the corresponding migration results before you migrate your existing configuration.

**Note** You cannot configure a mix of the old IPDT and IPv6 snooping CLI with the SISF-based device tracking CLI.

### Only IPDT Configuration Exists

If your device has only IPDT configuration, running the **device-tracking upgrade-cli** command converts the configuration to use the new SISF policy that is created and attached to the interface. You can then update this SISF policy.

If you continue to use the legacy commands, this restricts you to operate in a legacy mode where only the legacy IPDT and IPv6 snooping commands are available on the device.

### Only IPv6 Snooping Configuration Exists

On a device with existing IPv6 snooping configuration, the old IPv6 Snooping commands are available for further configuration. The following options are available:

- (Recommended) Use the **device-tracking upgrade-cli** command to convert all your legacy configuration to the new SISF-based device tracking commands. After conversion, only the new device tracking commands will work on your device.

- Use the legacy IPv6 Snooping commands for your future configuration and do not run the **device-tracking upgrade-cli** command. With this option, only the legacy IPv6 Snooping commands are available on your device, and you cannot use the new SISF-based device tracking CLI commands.

### Both IPDT and IPv6 Snooping Configuration Exist

On a device that has both legacy IPDT configuration and IPv6 snooping configuration, you can convert legacy commands to the SISF-based device tracking CLI commands. However, note that only one snooping policy can be attached to an interface, and the IPv6 snooping policy parameters override the IPDT settings.

**Note** If you do not migrate to the new SISF-based commands and continue to use the legacy IPv6 snooping or IPDT commands, your IPv4 device tracking configuration information may be displayed in the IPv6 snooping commands, as the SISF-based device tracking feature handles both IPv4 and IPv6 configuration. To avoid this, we recommend that you convert your legacy configuration to SISF-based device tracking commands.

### No IPDT or IPv6 Snooping Configuration Exists

If your device has no legacy IP Device Tracking or IPv6 Snooping configurations, you can use only the new SISF-based device tracking commands for all your future configuration. The legacy IPDT commands and IPv6 snooping commands are not available.

### Examples

The following example shows you how to convert IPv6 Snooping commands to SISF-based device-tracking commands.

```
Device# show ipv6 snooping features
Feature name   priority state
Device-tracking   128   READY
Source guard       32   READY

Device# configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# device-tracking upgrade-cli
 IPv6 Snooping and IPv4 device tracking CLI will be
 converted to the new top level device-tracking CLI
Are you sure ? [yes]: yes
Number of Snooping Policies Upgraded: 2
Device(config)# exit
```

After conversion, only the new SISF-based device-tracking commands will work on your device:

```
Device# show ipv6 snooping features
                          ^
% Invalid input detected at '^' marker.

Device# show device-tracking features
Feature name   priority state
Device-tracking   128   READY
Source guard       32   READY

Device# show device-tracking policies
Target               Type  Policy               Feature        Target range
```

```
Te1/0/4             PORT  sisf-01             Device-tracking vlan 200
vlan 200            VLAN  sisf-01             Device-tracking vlan all
```

# device sensor filter list mdns

To create a multicast DNS (mDNS) protocol filter containing a list of Type-Length-Value (TLV) fields that can be included or excluded in the device sensor output, use the **device-sensor filter-list mdns** command in global configuration mode. To remove the mDNS filter containing the list of TLV fields, use the **no** form of this command.

**device-sensor filter-list mdns list** *tlv-list-name*
**no device-sensor filter-list mdns list** *tlv-list-name*

## Syntax Description

| | |
|---|---|
| **list** | Specifies an mDNS TLV filter list. |
| *tlv-list-name* | Name of the mDNS TLV filter list. |

## Command Default

mDNS filter list is not available.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE 17.14.1 | The command was introduced. |

## Usage Guidelines

Use the **device-sensor filter-list mdns list** *tlv-list-name* command to configure the name of the mDNS TLV filter list and enter mDNS sensor configuration mode.

### Example

The following example shows how to create an mDNS filter containing a list of TLVs:

```
Device> enable
Device# configure terminal
Device(config)# device-sensor filter-list mdns list mdns-list
```

# device sensor filter spec

To apply a specific protocol filter containing a list of Type-Length-Value (TLV) fields to the device sensor output, use the **device-sensor filter-spec** command in global configuration mode. To remove the protocol filter list from the device sensor output, use the **no** form of this command.

**device-sensor filter-spec** { **cdp** | **dhcp** | **lldp** | **mdns** } { **exclude** { **all** | **list** *list-name* } | **include list** *list-name* }
`no device-sensor filter-spec { cdp | dhcp | lldp | mdns } { exclude { all | list` *list-name* `} | include list` *list-name* `}`

## Syntax Description

| | |
|---|---|
| **cdp** | Applies a Cisco Discovery Protocol TLV filter list to the device sensor output. |
| **dhcp** | Applies a DHCP TLV filter list to the device sensor output. |
| **lldp** | Applies a Link Layer Discovery Protocol (LLDP) TLV filter list to the device sensor output. |
| **mdns** | Applies a multicast DNS (mDNS) protocol TLV filter list to the device sensor output. |
| **exclude** | Specifies the TLVs that should be excluded from the device sensor output. |
| **all** | Disables all notifcations for the associated protocol. |
| **list** *list-name* | Specifies the name of the protocol TLV filter-list. |
| **include** | Specifies the TLVs that should be included in the device sensor output. |

## Command Default

All TLVs are included in notifications and will trigger notifications.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Fuji 16.8.1a | This command was introduced. |
| Cisco IOS XE 17.14.1 | The **mdns** option was introduced. |

## Usage Guidelines

Use the **device-sensor filter-spec** command to specify the TLVs that must be included in all sensor outputs (session notifications sent to internal sensor clients and accounting requests).

### Example

The following example shows how to apply a Cisco Discovery Protocol TLV filter list to the device sensor output:

```
Device> enable
Device# configure terminal
Device(config)# device-sensor filter-spec cdp include list cdp-list1
Device(config)# end
```

The following example shows how to apply a mDNS TLV filter list to the device sensor output:

```
Device> enable
Device# configure terminal
Device(config)# device-sensor filter-spec mdns include list mdns-list
Device(config)# end
```

# dnscrypt (Parameter Map)

To enable Domain Name System (DNS) packet encryption for authenticating communications between a Cisco device and the Cisco Umbrella Integration feature, use the **dnscrypt** command in parameter-map type inspect configuration mode. To disable DNS packet encryption, use the **no** form of this command.

**dnscrypt**
**no dnscrypt**

## Syntax Description

This command has no arguments or keywords.

## Command Default

DNS packet encyrption for umbrella mode is not configured.

## Command Modes

Parameter-map type inspect configuration (config-profile)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Usage Guidelines

When DNSCrypt is used, the DNS request packets' size is more than 512 bytes. Ensure that these packets are allowed through the intermediary devices; otherwise, the response may not reach the intended recipients.

## Examples

The following example shows how to enable DNS packet encryption:

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type umbrella global
Device(config-profile)# dnscrypt
```

## Related Commands

| Command | Description |
|---|---|
| **parameter-map type umbrella global** | Configures a parameter map type in umbrella mode. |

# dot1x authenticator eap profile

To configure the Extensible Authentication Protocol (EAP) profile to use during 802.1x authentication, use the **dot1x authenticator eap profile** command in interface configuration mode. To disable the EAP profile, use the **no** form of this command.

**dot1x authenticator eap profile** [*name*]
**no dot1x authenticator eap profile**

## Syntax Description

| | |
|---|---|
| *name* | EAP authenticator profile name. |

## Command Default

EAP profile is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Cupertino 17.7.1 | This command was introduced. |

## Usage Guidelines

You must enter the **switchport mode access** command on a switch port before entering this command.

The following example shows how to configure Cisco TrustSec manual configuration and 802.1x configurations together:

```
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# switchport mode access
Device(config-if)# cts manual
Device(config-if-cts-manual)# propagate sgt
Device(config-if-cts-manual)# policy static sgt 77 trusted
Device(config-if-cts-manual)# exit
Device(config-if)# dot1x pae authenticator
Device(config-if)# dot1x authenticator eap profile md5
```

## Related Commands

| Command | Description |
|---|---|
| **switchport mode access** | Sets the trunking mode to access m |

# dot1x critical (global configuration)

To configure the IEEE 802.1X critical authentication parameters, use the **dot1x critical** command in global configuration mode.

**dot1x critical eapol**

## Syntax Description

| | |
|---|---|
| **eapol** | Specifies that the switch send an EAPOL-Success message when the device successfully authenticates the critical port. |

## Command Default

**eapol** is disabled

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

This example shows how to specify that the device sends an EAPOL-Success message when the device successfully authenticates the critical port:

```
Device> enable
Device# configure terminal
Device(config)# dot1x critical eapol
Device(config)# exit
```

# dot1x logging verbose

To filter detailed information from 802.1x system messages, use the **dot1x logging verbose** command in global configuration mode on a device stack or on a standalone device.

**dot1x logging verbose**
**no dot1x logging verbose**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Detailed logging of system messages is not enabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

This command filters details, such as anticipated success, from 802.1x system messages. Failure messages are not filtered.

The following example shows how to filter verbose 802.1x system messages:

```
Device> enable
Device# configure terminal
Device(config)# dot1x logging verbose
Device(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **authentication logging verbose** | Filters details from authentication |
| **dot1x logging verbose** | Filters details from 802.1x system |
| **mab logging verbose** | Filters details from MAC authentic |

# dot1x max-start

To set the maximum number of Extensible Authentication Protocol over LAN (EAPOL) start frames that a supplicant sends (assuming that no response is received) to the client before concluding that the other end is 802.1X unaware, use the **dot1x max-start** command in interface configuration mode. To remove the maximum number-of-times setting, use the **no** form of this command.

**dot1x max-start** *number*
**no dot1x max-start**

## Syntax Description

| | |
|---|---|
| *number* | Maximum number of times that the router sends an EAPOL start frame. The value is from 1 to 10. The default is 3. |

## Command Default

The default maximum number setting is 3.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

You must enter the **switchport mode access** command on a switch port before entering this command.

The following example shows that the maximum number of EAPOL Start requests has been set to 5:

```
Device> enable
Device# configure terminal
Device(config)# interface gigibitethernet 1/0/3
Device(config-if)# dot1x max-start 5
Device(config-if)# end
```

# dot1x pae

To set the Port Access Entity (PAE) type, use the **dot1x pae** command in interface configuration mode. To disable the PAE type that was set, use the **no** form of this command.

**dot1x pae** {**supplicant** | **authenticator**}
**no dot1x pae** {**supplicant** | **authenticator**}

## Syntax Description

| | |
|---|---|
| **supplicant** | The interface acts only as a supplicant and will not respond to messages that are meant for an authenticator. |
| **authenticator** | The interface acts only as an authenticator and will not respond to any messages meant for a supplicant. |

## Command Default

PAE type is not set.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **no dot1x pae** interface configuration command to disable IEEE 802.1x authentication on the port.

When you configure IEEE 802.1x authentication on a port, such as by entering the **dot1x port-control** interface configuration command, the device automatically configures the port as an IEEE 802.1x authenticator. After the **no dot1x pae** interface configuration command is entered, the Authenticator PAE operation is disabled.

The following example shows that the interface has been set to act as a supplicant:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/3
Device(config-if)# dot1x pae supplicant
Device(config-if)# end
```

# dot1x supplicant controlled transient

To control access to an 802.1x supplicant port during authentication, use the **dot1x supplicant controlled transient** command in global configuration mode. To open the supplicant port during authentication, use the **no** form of this command

**dot1x supplicant controlled transient**
**no dot1x supplicant controlled transient**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Access is allowed to 802.1x supplicant ports during authentication.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

In the default state, when you connect a supplicant device to an authenticator switch that has BPCU guard enabled, the authenticator port could be error-disabled if it receives a Spanning Tree Protocol (STP) bridge protocol data unit (BPDU) packets before the supplicant switch has authenticated. You can control traffic exiting the supplicant port during the authentication period. Entering the **dot1x supplicant controlled transient** command temporarily blocks the supplicant port during authentication to ensure that the authenticator port does not shut down before authentication completes. If authentication fails, the supplicant port opens. Entering the **no dot1x supplicant controlled transient** command opens the supplicant port during the authentication period. This is the default behavior.

We recommend using the **dot1x supplicant controlled transient** command on a supplicant device when BPDU guard is enabled on the authenticator switch port with the **spanning-tree bpduguard enable** interface configuration command.

This example shows how to control access to 802.1x supplicant ports on a device during authentication:

```
Device> enable
Device# configure terminal
Device(config)# dot1x supplicant controlled transient
Device(config)# exit
```

# dot1x supplicant force-multicast

To force a supplicant switch to send only multicast Extensible Authentication Protocol over LAN (EAPOL) packets whenever it receives multicast or unicast EAPOL packets, use the **dot1x supplicant force-multicast** command in global configuration mode. To return to the default setting, use the **no** form of this command.

**dot1x supplicant force-multicast**
**no dot1x supplicant force-multicast**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The supplicant device sends unicast EAPOL packets when it receives unicast EAPOL packets. Similarly, it sends multicast EAPOL packets when it receives multicast EAPOL packets.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Enable this command on the supplicant device for Network Edge Access Topology (NEAT) to work in all host modes.

This example shows how force a supplicant device to send multicast EAPOL packets to the authenticator device:

```
Device> enable
Device# configure terminal
Device(config)# dot1x supplicant force-multicast
Device(config)# end
```

## Related Commands

| Command | Description |
|---|---|
| **cisp enable** | Enables CISP on a device so that it |
| **dot1x credentials** | Configures the 802.1x supplicant c |
| **dot1x pae supplicant** | Configures an interface to act only |

# dot1x test eapol-capable

To monitor IEEE 802.1x activity on all the switch ports and to display information about the devices that are connected to the ports that support IEEE 802.1x, use the **dot1x test eapol-capable** command in privileged EXEC mode.

**dot1x test eapol-capable** [**interface** *interface-id*]

## Syntax Description

| | |
|---|---|
| **interface** *interface-id* | (Optional) Port to be queried. |

## Command Default

There is no default setting.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use this command to test the IEEE 802.1x capability of the devices connected to all ports or to specific ports on a switch.

There is not a no form of this command.

This example shows how to enable the IEEE 802.1x readiness check on a switch to query a port. It also shows the response received from the queried port verifying that the device connected to it is IEEE 802.1x-capable:

```
Device> enable
Device# dot1x test eapol-capable interface gigabitethernet1/0/13

DOT1X_PORT_EAPOL_CAPABLE:DOT1X: MAC 00-01-02-4b-f1-a3 on gigabitethernet1/0/13 is EAPOL
capable
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x test timeout** *timeout* | Configures the timeout used to w<br>readiness query. |

# dot1x test timeout

To configure the timeout used to wait for EAPOL response from a port being queried for IEEE 802.1x readiness, use the **dot1x test timeout** command in global configuration mode.

**dot1x test timeout** *timeout*

## Syntax Description

| | |
|---|---|
| *timeout* | Time in seconds to wait for an EAPOL response. The range is from 1 to 65535 seconds. |

## Command Default

The default setting is 10 seconds.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use this command to configure the timeout used to wait for EAPOL response.

There is not a no form of this command.

This example shows how to configure the switch to wait 27 seconds for an EAPOL response:

```
Device> enable
Device# dot1x test timeout 27
```

You can verify the timeout configuration status by entering the **show running-config** command.

## Related Commands

| Command | Description |
|---|---|
| **dot1x test eapol-capable** [**interface** *interface-id*] | Checks for IEEE 802.1x readiness on devices connected to all or to specified IEEE 802.1x-capable ports. |

# dot1x timeout

To configure the value for retry timeouts, use the **dot1x timeout** command in global configuration or interface configuration mode. To return to the default value for retry timeouts, use the **no** form of this command.

**dot1x timeout** { **auth-period** *seconds* | **held-period** *seconds* | **quiet-period** *seconds* | **ratelimit-period** *seconds* | **server-timeout** *seconds* | **start-period** *seconds* | **supp-timeout** *seconds* | **tx-period** *seconds* }

## Syntax Description

| | |
|---|---|
| **auth-period** *seconds* | Configures the time, in seconds for which a supplicant will stay in the HELD state (that is, the length of time it will wait before trying to send the credentials again after a failed attempt).<br>The range is from 1 to 65535. The default is 30. |
| **held-period** *seconds* | Configures the time, in seconds for which a supplicant will stay in the HELD state (that is, the length of time it will wait before trying to send the credentials again after a failed attempt).<br>The range is from 1 to 65535. The default is 60 |
| **quiet-period** *seconds* | Configures the time, in seconds, that the authenticator (server) remains quiet (in the HELD state) following a failed authentication exchange before trying to reauthenticate the client.<br>The range is from 1 to 65535. The default is 60 |
| **ratelimit-period** *seconds* | Throttles the EAP-START packets that are sent from misbehaving client PCs (for example, PCs that send EAP-START packets that result in the wasting of device processing power).<br>• The authenticator ignores EAPOL-Start packets from clients that have successfully authenticated for the rate-limit period duration.<br>• The range is from 1 to 65535. By default, rate limiting is disabled. |
| **server-timeout** *seconds* | Configures the interval, in seconds, between two successive EAPOL-Start frames when they are being retransmitted.<br>• The range is from 1 to 65535. The default is 30.<br>If the server does not send a response to an 802.1X packet within the specified period, the packet is sent again. |
| **start-period** *seconds* | Configures the interval, in seconds, between two successive EAPOL-Start frames when they are being retransmitted.<br>The range is from 1 to 65535. The default is 30. |

| | |
|---|---|
| **supp-timeout** *seconds* | Sets the authenticator-to-supplicant retransmission time for all EAP messages other than EAP Request ID.<br>The range is from 1 to 65535. The default is 30. |
| **tx-period** *seconds* | Configures the number of seconds between retransmission of EAP request ID packets (assuming that no response is received) to the client.<br>• The range is from 1 to 65535. The default is 30.<br>• If an 802.1X packet is sent to the supplicant and the supplicant does not send a response after the retry period, the packet will be sent again. |

**Command Default**

Periodic reauthentication and periodic rate-limiting are done.

**Command Modes**

Global configuration (config)

Interface configuration (config-if)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

**Usage Guidelines**

You should change the default value of this command only to adjust for unusual circumstances such as unreliable links or specific behavioral problems with certain clients and authentication servers.

The **dot1x timeout reauth-period** interface configuration command affects the behavior of the device only if you have enabled periodic re-authentication by using the **dot1x reauthentication** interface configuration command.

During the quiet period, the device does not accept or initiate any authentication requests. If you want to provide a faster response time to the user, enter a number smaller than the default.

When the **ratelimit-period** is set to 0 (the default), the device does not ignore EAPOL packets from clients that have been successfully authenticated and forwards them to the RADIUS server.

The following example shows that various 802.1X retransmission and timeout periods have been set:

```
Device> enable
Device(config)# configure terminal
Device(config)# interface gigabitethernet 1/0/3
Device(config-if)# dot1x port-control auto
Device(config-if)# dot1x timeout auth-period 2000
Device(config-if)# dot1x timeout held-period 2400
Device(config-if)# dot1x timeout quiet-period 600
Device(config-if)# dot1x timeout start-period 90
Device(config-if)# dot1x timeout supp-timeout 300
Device(config-if)# dot1x timeout tx-period 60
Device(config-if)# dot1x timeout server-timeout 60
Device(config-if)# end
```

# dscp

To configure DSCP marking for authentication and accounting on RADIUS packets, use the **dscp** command. To disable DSCP marking for authentication and accounting on RADIUS packets, use the **no** form of this command

**dscp** { **acct** *dscp_acct_value* | **auth** *dscp_auth_value* }

**no dscp** { **acct** *dscp_acct_value* | **auth** *dscp_auth_value* }

## Syntax Description

| | |
|---|---|
| **acct** *dscp_acct_value* | Configures RADIUS DSCP marking value for accounting. The valid range is from 1 to 63. The default value is 0. |
| **auth** *dscp_auth_value* | Configures RADIUS DSCP marking value for authentication. The valid range is from 1 to 63. The default value is 0. |

## Command Default

The DSCP marking on RADIUS packets is disabled by default.

## Command Modes

RADIUS server configuration (config-radius-server)
RADIUS server group configuration (config-sg-radius)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Bengaluru 17.5.1 | This command was introduced. |

### Example

This example shows how to configure DSCP marking for authentication and accounting on RADIUS packets for a RADIUS server:

```
Device(config)#radius server abc
Device(config-radius-server)#address ipv4 10.1.1.1 auth-port 1645 acct-port 1646
Device(config-radius-server)#dscp auth 10 acct 20
Device(config-radius-server)#key cisco123
Device(config-radius-server)#end
```

This example shows how to configure DSCP marking for authentication and accounting on RADIUS packets for a RADIUS server group:

```
Device(config)#aaa group server radius xyz
Device(config-sg-radius)#server name abc
Device(config-sg-radius)#ip radius source-interface Vlan18
Device(config-sg-radius)#dscp auth 30 acct 10
Device(config-sg-radius)#end
```

# dtls

To configure Datagram Transport Layer Security (DTLS) parameters, use the **dtls** command in radius server configuration mode. To return to the default setting, use the **no** form of this command.

**dtls** [ **connectiontimeout** *connection-timeout-value* | **idletimeout** *idle-timeout-value* | [ **ip** | **ipv6** ] { **radius source-interface** *interface-name* | **vrf forwarding** *forwarding-table-name* } | **match-server-identity** { **email-address** *email-address* | **hostname** *hostname* | **ip-address** *ip-address* } | **port** *port-number* | **retries** *number-of-connection-retries* | **trustpoint** { **client** *trustpoint name* | **server** *trustpoint name* } ]

**no dtls**

## Syntax Description

| | |
|---|---|
| **connectiontimeout** *connection-timeout-value* | (Optional) Configures the DTLS connection timeout value. |
| **idletimeout** *idle-timeout-value* | (Optional) Configures the DTLS idle timeout value. |
| [**ip** \| **ipv6**] {**radius source-interface** *interface-name* \| **vrf forwarding** *forwarding-table-name*} | (Optional) Configures IP or IPv6 source parameters. |
| **match-server-identity** {**email-address** *email-address* \| **hostname** *host-name* \| **ip-address** *ip-address*} | Configures RadSec certification validation parameters. |
| **port** *port-number* | (Optional) Configures the DTLS port number. |
| **retries** *number-of-connection-retries* | (Optional) Configures the number of DTLS connection retries. |
| **trustpoint** {**client** *trustpoint name* \| **server** *trustpoint name*} | (Optional) Configures the DTLS trustpoint for the client and the server. |

## Command Default

- The default value of DTLS connection timeout is 5 seconds.
- The default value of DTLS idle timeout is 60 seconds.
- The default DTLS port number is 2083.
- The default value of DTLS connection retries is 5.

## Command Modes

Radius server configuration (config-radius-server)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.6.1 | This command was introduced. |
| Cisco IOS XE Gibraltar 16.10.1 | The **match-server-identity** keyword was introduced. |
| Cisco IOS XE Amsterdam 17.1.1 | The **ipv6** keyword was introduced. |

## Usage Guidelines

We recommend that you use the same server type, either only Transport Layer Security (TLS) or only DTLS, under an Authentication, Authorization, and Accounting (AAA) server group.

## Examples

The following example shows how to configure the DTLS connection timeout value to 10 seconds:

```
Device> enable
Device# configure terminal
Device(config)# radius server R1
Device(config-radius-server)# dtls connectiontimeout 10
Device(config-radius-server)# end
```

## Related Commands

| Command | Description |
|---|---|
| **show aaa servers** | Displays information related to the DTLS server. |
| **clear aaa counters servers radius** | Clears the RADIUS DTLS-specific statistics. |
| **debug radius dtls** | Enables RADIUS DTLS-specific debugs. |

# enable password

To set a local password to control access to various privilege levels, use the **enable password** command in global configuration mode. To remove control access of the local password, use the **no** form of this command.

**enable** [ **common-criteria-policy** *policy-name* ] **password** [ **level** *level* ] { [**0**] *unencrypted-password* | [ *encryption-type*] *encrypted-password* }
**no enable** [ **common-criteria-policy** *policy-name* ] **password** [ **level** *level* ]

## Syntax Description

| | |
|---|---|
| **common-criteria-policy** *policy-name* | (Optional) Specifies a AAA common criteria policy name. |
| **level** *level* | (Optional) Specifies the level for which the password is applicable. You can sp levels, using numbers 0 through 15. Level 1 is normal user EXEC mode user p specified in the command or in the **no** form of the command, the privilege leve |
| **0** | (Optional) Specifies an unencrypted cleartext password. The password is conve Algorithm (SHA) 256 secret and is stored in the device. |
| *unencrypted-password* | Specifies the password to enter enable mode. |
| *encryption-type* | (Optional) Cisco-proprietary algorithm used to encrypt the password. If you spe next argument that you supply must be an encrypted password (a password alrea device). You can specify type 7, which indicates that a hidden password follow |
| *encrypted-password* | Encrypted password copied from another device configuration. |

## Command Default

No password is defined.

## Command Modes

Global configuration (config)

## Command History

| Release | Modif |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This c |
| Cisco IOS XE Cupertino 17.8.1 | The c<br>The **c**<br>keywo |

## Usage Guidelines

For the **common-criteria-policy** option, specify a policy name defined using the **aaa common-criteria policy** command. If you select this option, the password must be set based on the criteria defined in that particular AAA common criteria policy.

**Note**

- The **aaa new-model** and **aaa common-criteria policy** commands must be configured before attaching the **common-criteria-policy** option to the password.
- The **common-criteria-policy** option is not supported for the **enable secret** command.

If neither the **enable password** command nor the **enable secret** command is configured, and if a line password is configured for the console, the console line password serves as the enable password for all VTY (Telnet and Secure Shell [SSH]) sessions.

Use the **enable password** command with the **level** option to define a password for a specific privilege level. After you specify the level and the password, share the password with users who need to access this level. Use the **privilege level** configuration command to specify the commands that are accessible at various levels.

Typically, you enter an encryption type only if you copy and paste a password that has already been encrypted by a Cisco device, into this command.

**Caution** If you specify an encryption type and then enter a cleartext password, you will not be able to re-enter enable mode. You cannot recover a lost password that has been encrypted earlier.

If the **service password-encryption** command is set, the encrypted form of the password you create with the **enable password** command is displayed when the **more nvram:startup-config** command is run.

You can enable or disable password encryption with the **service password-encryption** command.

An enable password is defined as follows:

- Must contain a combination of numerals from 1 to 25, and uppercase and lowercase alphanumeric characters.
- Can have leading spaces, but they are ignored. However, intermediate and trailing spaces are recognized.
- Can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when you create the password, for example, to create the password *abc?123,* do the following:
  1. Enter **abc**.
  2. Press **Crtl-v**.
  3. Enter **?123**.

**Note** When the system prompts you to enter the **enable password** command, you need not precede the question mark with Ctrl-V; you can enter **abc?123** at the password prompt.

## Examples

The following example shows how to enables the password pswd2 for privilege level 2:

```
Device> enable
Device# configure terminal
Device(config)# enable password level 2 pswd2
```

The following example shows how to set the encrypted password $1$i5Rkls3LoyxzS8t9, which has been copied from a device configuration file, for privilege level 2 using encryption type 7:

```
Device> enable
Device# configure terminal
Device(config)# enable password level 2 5 $1$i5Rkls3LoyxzS8t9
```

**Related Commands**

| Command | Description |
|---|---|
| **enable secret** | Specifies an additional layer of security over the **enable** |
| **more nvram:startup-config** | Displays the startup configuration file contained in NVR CONFIG_FILE environment variable. |
| **privilege level** | Sets the privilege level for the user. |
| **service password-encryption** | Encrypts a password. |

# enable secret

To specify an additional layer of security over the **enable password** command, use the **enable secret** command in global configuration mode. To turn off the enable secret function, use the **no** form of this command.

**enable secret** [**level** *level*] {[**0**] *unencrypted-password* | *encryption-type encrypted-password*}
**no enable secret** [**level** *level*] [*encryption-type encrypted-password*]

## Syntax Description

| | |
|---|---|
| **level** *level* | (Optional) Specifies the level for which the password is applicable. You can specify levels, using numerals 1 through 15. Level 1 is normal user EXEC mode privileges. If in the command or in the **no** form of the command, the privilege level defaults to 15 |
| **0** | (Optional) Specifies an unencrypted cleartext password. The password is converted Algorithm (SHA) 256 secret and is stored in the device. |
| *unencrypted-password* | Specifies the password for users to enter enable mode. This password should be differe created with the **enable password** command. |
| *encryption-type* | Cisco-proprietary algorithm used to hash the password:<br>• **5**: Specifies a message digest algorithm 5-encrypted (MD5-encrypted) secret.<br>• **8**: Specifies a Password-Based Key Derivation Function 2 (PBKDF2) with SHA<br>• **9**: Specifies a scrypt-hashed secret. |
| *encrypted-password* | Hashed password that is copied from another device configuration. |

## Command Default

No password is defined.

## Command Modes

Global configuration (config)

## Command History

| Release | Modificat |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This comm |

## Usage Guidelines

If neither the **enable password** command or the **enable secret** command is configured, and if a line password is configured for the console, the console line password serves as the enable password for all vty (Telnet and Secure Shell [SSH]) sessions.

Use the **enable secret** command to provide an additional layer of security over the **enable password** password. The **enable secret** command provides better security by storing the password using a nonreversible cryptographic function. The additional layer of security encryption is useful in environments where the password is sent to the network or is stored on a TFTP server.

Typically, you enter an encryption type only when you paste an encrypted password that you copied from a device configuration file, into this command.

**Caution** If you specify an encryption type and then enter a cleartext password, you will not be able to reenter enable mode. You cannot recover a lost password that has been encrypted earlier.

If you use the same password for the **enable password** and **enable secret** commands, you receive an error message warning that this practice is not recommended, but the password will be accepted. By using the same password, however, you undermine the additional security the **enable secret** command provides.

**Note** After you set a password using the **enable secret** command, a password set using the **enable password** command works only if the **enable secret** is disabled. Additionally, you cannot recover a lost password that has been encrypted by any method.

If the **service password-encryption** command is set, the encrypted form of the password you create is displayed when the **more nvram:startup-config** command is run.

You can enable or disable password encryption with the **service password-encryption** command.

An enable password is defined as follows:

- Must contain a combination of numerals from 1 to 25, and uppercase and lowercase alphanumeric characters.
- Can have leading spaces, but they are ignored. However, intermediate and trailing spaces are recognized.
- Can contain the question mark (?) character if you precede the question mark with the key combination Crtl-v when you create the password; for example, to create the password *abc?123,* do the following:
  1. Enter **abc**.
  2. Press **Crtl-v**.
  3. Enter **?123**.

**Note** When the system prompts you to enter the **enable password** command, you need not precede the question mark with Ctrl-v; you can enter **abc?123** at the password prompt.

## Examples

The following example shows how to specify a password with the **enable secret** command:

```
Device> enable
Device# configure terminal
Device(config)# enable secret password
```

After specifying a password with the **enable secret** command, users must enter this password to gain access. Otherwise, passwords set using the **enable password** command will no longer work.

```
Password: password
```

The following example shows how to enable the encrypted password $1$FaD0$Xyti5Rkls3LoyxzS8, which has been copied from a device configuration file, for privilege level 2, using the encryption type 4:

```
Device> enable
Device# configure terminal
Device(config)# enable password level 2 4 $1$FaD0$Xyti5Rkls3LoyxzS8
```

The following example shows the warning message that is displayed when a user enters the **enable secret 4** *encrypted-password* command:

```
Device> enable
Device# configure terminal
Device(config)# enable secret 4 tnhtc92DXBhelxjYk8LWJrPV36S2i4ntXrpb4RFmfqY

WARNING: Command has been added to the configuration but Type 4 passwords have been
deprecated.
Migrate to a supported password type

Device(config)# end
Device# show running-config | inc secret

enable secret 4 tnhtc92DXBhelxjYk8LWJrPV36S2i4ntXrpb4RFmfqY
```

## Related Commands

| Command | Description |
|---|---|
| **enable password** | Sets a local password to control access to various privilege |
| **more nvram:startup-config** | Displays the startup configuration file contained in NVRAM CONFIG_FILE environment variable. |
| **service password-encryption** | Encrypt passwords. |

# epm access-control open

To configure an open directive for ports that do not have an access control list (ACL) configured, use the **epm access-control open** command in global configuration mode. To disable the open directive, use the **no** form of this command.

**epm access-control open**
**no epm access-control open**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The default directive applies.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use this command to configure an open directive that allows hosts without an authorization policy to access ports configured with a static ACL. If you do not configure this command, the port applies the policies of the configured ACL to the traffic. If no static ACL is configured on a port, both the default and open directives allow access to the port.

You can verify your settings by entering the **show running-config** command.

This example shows how to configure an open directive.

```
Device> enable
Device# configure terminal
Device(config)# epm access-control open
Device(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **show running-config** | Displays the contents of the current running configuration file. |

# fqdn ttl-timeout factor

To configure the time-to-live (TTL) timeout factor of a FQDN local cache entry, use the **fqdn ttl-timeout factor** command in global configuration mode. To remove the TTL timeout factor, use the **no** form of this command.

**fqdn ttl-timeout factor** { *ttl-timeout* | **infinite** }

**no fqdn ttl-timeout factor** { *ttl-timeout* | **infinite** }

## Syntax Description

| | |
|---|---|
| *ttl-timeout* | Specifies the TTL timeout factor. The range is from 1 to 300. |
| **infinite** | The FQDN entry stays forever. |

## Command Default

The default TTL timeout factor is 10.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Bengaluru 17.5.1 | The command was introduced. |

## Usage Guidelines

The local TTL value for the cached FQDN entry is calculated based on the TTL value received in the DNS response packet multiplied by the configured TTL timeout factor. For example, if the received DNS response has a TTL value of 100 seconds and multiplication factor is set to 10, then the local TTL for the cached FQDN entry will be 1000 seconds.

## Examples

The following example shows how to set the FQDN TTL timeout factor:

```
Device> enable
Device# configure terminal
Device(config)# fqdn ttl-timeout-factor 100
```

# include-icv-indicator

To include the integrity check value (ICV) indicator in MKPDU, use the **include-icv-indicator** command in MKA-policy configuration mode. To disable the ICV indicator, use the **no** form of this command.

**include-icv-indicator**
**no include-icv-indicator**

## Syntax Description

This command has no arguments or keywords.

## Command Default

ICV indicator is included.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Examples

The following example shows how to include the ICV indicator in MKPDU:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# include-icv-indicator
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **confidentiality-offset** | Sets the confidentiality offset for MACsec operations. |
| **delay-protection** | Configures MKA to use delay protection in sending MKPDU. |
| **key-server** | Configures MKA key-server options. |
| **macsec-cipher-suite** | Configures cipher suite for deriving SAK. |
| **sak-rekey** | Configures the SAK rekey interval. |
| **send-secure-announcements** | Configures MKA to send secure announcements in sending MKPDUs. |
| **ssci-based-on-sci** | Computes SSCI based on the SCI. |
| **use-updated-eth-header** | Uses the updated Ethernet header for ICV calculation. |

# ip access-list

To define an IP access list or object-group access control list (ACL) by name or number or to enable filtering for packets with IP helper-address destinations, use the **ip access-list** command in global configuration mode. To remove the IP access list or object-group ACL or to disable filtering for packets with IP helper-address destinations, use the **no** form of this command.

**ip access-list** { {**extended** | **resequence** | **standard**} {*access-list-numberaccess-list-name*} | **helper egress check** | **log-update threshold** *threshold-number* | **logging** {**hash-generation** | **interval** *time*} | **persistent** | **role-based** *access-list-name* | **fqdn** *access-list-name* }
**no ip access-list** { { **extended** | **resequence** | **standard** } { *access-list-number access-list-name* } | **helper egress check** | **log-update threshold** | **logging** { **hash-generation** | **interval** } | **persistent** | **role-based** *access-list-name* | **fqdn** *access-list-name* }

## Syntax Description

| | |
|---|---|
| **standard** | Specifies a standard IP access list. |
| **resequence** | Specifies a resequenced IP access list. |
| **extended** | Specifies an extended IP access list. Required for object-group ACLs. |
| *access-list-name* | Name of the IP access list or object-group ACL. Names cannot contain a space or quotation mark, and must begin with an alphabetic character to prevent ambiguity with numbered access lists. |
| *access-list-number* | Number of the access list.<br>• A standard IP access list is in the ranges 1-99 or 1300-1999.<br>• An extended IP access list is in the ranges 100-199 or 2000-2699. |
| **helper egress check** | Enables permit or deny matching capability for an outbound access list that is applied to an interface, for traffic that is relayed via the IP helper feature to a destination server address. |
| **log-update** | Controls the access list log updates. |
| **threshold** *threshold-number* | Sets the access list logging threshold. The range is 0 to 2147483647. |
| **logging** | Controls the access list logging. |
| **hash-generation** | Enables syslog hash code generation. |
| **interval** *time* | Sets the access list logging interval in milliseconds. The range is 0 to 2147483647. |
| **persistent** | Access control entry (ACE) sequence numbers are persistent across reloads.<br>**Note**<br>This is enabled by default and cannot be disabled. |
| **role-based** | Specifies a role-based IP access list. |

| | |
|---|---|
| **fqdn** | Specifies a FQDN IP access list.<br>**Note**<br>The name must start with an alphabet. |

**Command Default**

No IP access list or object-group ACL is defined, and outbound ACLs do not match and filter IP helper relayed traffic.

**Command Modes**

Global configuration (config)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Bengaluru 17.4.1 | The **fqdn** keyword was introduced. |

**Usage Guidelines**

Use this command to configure a named or numbered IP access list or an object-group ACL. This command places the device in access-list configuration mode, where you must define the denied or permitted access conditions by using the **deny** and **permit** commands.

Specifying the **standard** or **extended** or **fqdn** keyword with the **ip access-list** command determines the prompt that appears when you enter access-list configuration mode. You must use the **extended** keyword when defining object-group ACLs.

You can create object groups and IP access lists or object-group ACLs independently, which means that you can use object-group names that do not yet exist.

Use the **ip access-group** command to apply the access list to an interface.

The **ip access-list helper egress check** command enables outbound ACL matching for permit or deny capability on packets with IP helper-address destinations. When you use an outbound extended ACL with this command, you can permit or deny IP helper relayed traffic based on source or destination User Datagram Protocol (UDP) ports. The **ip access-list helper egress check** command is disabled by default; outbound ACLs will not match and filter IP helper relayed traffic.

**Examples**

The following example defines a standard access list named Internetfilter:

```
Device> enable
Device# configure terminal
Device(config)# ip access-list standard Internetfilter
Device(config-std-nacl)# permit 192.168.255.0 0.0.0.255
Device(config-std-nacl)# permit 10.88.0.0 0.0.255.255
Device(config-std-nacl)# permit 10.0.0.0 0.255.255.255
```

The following example shows how to set the FQDN TTL timeout factor and create an FQDN ACL named facl.

```
Device> enable
Device# configure terminal
Device(config)# fqdn ttl-timeout-factor 100
Device(config)# ip access-list fqdn facl
Device(config-fqdn-acl)# 100 permit ip any any
Device(config-fqdn-acl)# 10 permit ip host 192.0.2.121 host dynamic www.google.com
Device(config-fqdn-acl)# end
```

The following example shows how to create an object-group ACL that permits packets from the users in my_network_object_group if the protocol ports match the ports specified in my_service_object_group:

```
Device> enable
Device# configure terminal
Device(config)# ip access-list extended my_ogacl_policy
Device(config-ext-nacl)# permit tcp object-group my_network_object_group portgroup
 my_service_object_group any
Device(config-ext-nacl)# deny tcp any any
```

The following example shows how to enable outbound ACL filtering on packets with helper-address destinations:

```
Device> enable
Device# configure terminal
Device(config)# ip access-list helper egress check
```

## Related Commands

| Command | Description |
|---|---|
| **deny** | Sets conditions in a named IP access list or in an object-group ACL that will deny packets. |
| **ip access-group** | Applies an ACL or an object-group ACL to an interface or a service policy map. |
| **object-group network** | Defines network object groups for use in object-group ACLs. |
| **object-group service** | Defines service object groups for use in object-group ACLs. |
| **permit** | Sets conditions in a named IP access list or in an object-group ACL that will permit packets. |
| **show ip access-list** | Displays the contents of IP access lists or object-group ACLs. |
| **show object-group** | Displays information about object groups that are configured. |

# ip access-list role-based

To create a role-based (security group) access control list (RBACL) and enter role-based ACL configuration mode, use the **ip access-list role-based** command in global configuration mode. To remove the configuration, use the **no** form of this command.

**ip access-list role-based** *access-list-name*
**no ip access-list role-based** *access-list-name*

## Syntax Description

| | |
|---|---|
| *access-list-name* | Name of the security group access control list (SGACL). |

## Command Default

Role-based ACLs are not configured.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

For SGACL logging, you must configure the **permit ip log** command. Also, this command must be configured in Cisco Identity Services Engine (ISE) to enable logging for dynamic SGACLs.

The following example shows how to define an SGACL that can be applied to IPv4 traffic and enter role-based access list configuration mode:

```
Device> enable
Device# configure terminal
Device(config)# ip access-list role-based rbacl1
Device(config-rb-acl)# permit ip log
Device(config-rb-acl)# end
```

## Related Commands

| Command | Description |
|---|---|
| **permit ip log** | Permits logging that matches the configured entry. |
| **show ip access-list** | Displays contents of all current IP access lists. |

# ip admission

To enable web authentication, use the **ip admission** command in interface configuration mode or fallback-profile configuration mode. To disable web authentication, use the **no** form of this command.

**ip admission** *rule*
**no ip admission** *rule*

## Syntax Description

| | |
|---|---|
| *rule* | IP admission rule name. |

## Command Default

Web authentication is disabled.

## Command Modes

Interface configuration (config-if)

Fallback-profile configuration (config-fallback-profile)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **ip admission** command applies a web authentication rule to a switch port.

This example shows how to apply a web authentication rule to a switchport:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip admission rule1
Device(config-if)# end
```

This example shows how to apply a web authentication rule to a fallback profile for use on an IEEE 802.1x enabled switch port.

```
Device> enable
Device# configure terminal
Device(config)# fallback profile profile1
Device(config-fallback-profile)# ip admission rule1
Device(config-fallback-profile)# end
```

# ip admission name

To enable web authentication, use the **ip admission name** command in global configuration mode. To disable web authentication, use the **no** form of this command.

**ip admission name** *name* {**consent** | **proxy http**} [**absolute timer** *minutes* | **inactivity-time** *minutes* | **list** {*acl* | *acl-name*} | **service-policy type tag** *service-policy-name*]
**no ip admission name** *name* {**consent** | **proxy http**} [**absolute timer** *minutes* | **inactivity-time** *minutes* | **list** {*acl* | *acl-name*} | **service-policy type tag** *service-policy-name*]

## Syntax Description

| | |
|---|---|
| *name* | Name of network admission control rule. |
| **consent** | Associates an authentication proxy consent web page with the IP admission rule specified using the *admission-name* argument. |
| **proxy http** | Configures web authentication custom page. |
| **absolute-timer** *minutes* | (Optional) Elapsed time, in minutes, before the external server times out. |
| **inactivity-time** *minutes* | (Optional) Elapsed time, in minutes, before the external file server is deemed unreachable. |
| **list** | (Optional) Associates the named rule with an access control list (ACL). |
| *acl* | Applies a standard, extended list to a named admission control rule. The value ranges from 1 through 199, or from 1300 through 2699 for expanded range. |
| *acl-name* | Applies a named access list to a named admission control rule. |
| **service-policy type tag** | (Optional) A control plane service policy is to be configured. |
| *service-policy-name* | Control plane tag service policy that is configured using the **policy-map type control tag***policyname* command, keyword, and argument. This policy map is used to apply the actions on the host when a tag is received. |

## Command Default

Web authentication is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **ip admission name** command globally enables web authentication on a switch.

After you enable web authentication on a switch, use the **ip access-group in** and **ip admission web-rule** interface configuration commands to enable web authentication on a specific interface.

## Examples

This example shows how to configure only web authentication on a switch port:

```
Device> enable
Device# configure terminal
Device(config) ip admission name http-rule proxy http
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 101 in
Device(config-if)# ip admission rule
Device(config-if)# end
```

This example shows how to configure IEEE 802.1x authentication with web authentication as a fallback mechanism on a switch port:

```
Device> enable
Device# configure terminal
Device(config)# ip admission name rule2 proxy http
Device(config)# fallback profile profile1
Device(config)# ip access group 101 in
Device(config)# ip admission name rule2
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# dot1x port-control auto
Device(config-if)# dot1x fallback profile1
Device(config-if)# end
```

## Related Commands

| Command | Description |
|---|---|
| **dot1x fallback** | Configures a port to use web authentication as a fallback method for clients that do not support IEEE 802.1x authentication. |
| **fallback profile** | Creates a web authentication fallback profile. |
| **ip admission** | Enables web authentication on a port. |
| **show authentication sessions interface** *interface* **detail** | Displays information about the web authentication session status. |
| **show ip admission** | Displays information about NAC cached entries or the NAC configuration. |

# ip dhcp restrict-next-hop

To assign DHCP IP address only to the neighboring device in an interface, use the **ip dhcp restrict-next-hop** command in interface configuration mode. To disable the feature, use the **no** form of the command.

**ip dhcp restrict-next-hop** { **both** | **cdp** | **lldp** }

## Syntax Description

| | |
|---|---|
| **both** | Restricts DHCP lease to both LLDP and CDP neighbors. |
| **cdp** | Restricts DHCP lease to CDP neighbors. |
| **lldp** | Restricts DHCP lease to LLDP neighbors. |

## Command Default

No default behavior.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Dublin 17.12.1 | This command was introduced. |

## Usage Guidelines

When the command is enabled, the DHCP server in the interface uses the mac addresses in the DHCP packet and compares it with the addresses in the CDP or LLDP cache table. If the mac addresses match, then the DHCP IP address is assigned to that device. If the mac addresses do not match, then the DHCP request is declined.

- The command is only supported if CDP or LLDP protocols are enabled on the interface.
- The command is not supported on stack setups and high availability devices.
- The command is not supported on port channels and SVI.

### Example

The following example shows how to assign DHCP IP address to both CDP and LLDP neighbors in an interface:

```
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip dhcp restrict-next-hop both
```

The following example shows how to assign DHCP IP address to only CDP neighbors in an interface:

```
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# ip dhcp restrict-next-hop cdp
```

The following example shows how to assign DHCP IP address to only LLDP neighbors in an interface:

```
Device(config)# interface gigabitethernet1/0/3
Device(config-if)# ip dhcp restrict-next-hop LLDP
```

# ip dhcp snooping database

To configure the Dynamic Host Configuration Protocol (DHCP)-snooping database, use the **ip dhcp snooping database** command in global configuration mode. To disable the DHCP-snooping database, use the **no** form of this command.

**ip dhcp snooping database** { **crashinfo:** *url* | **flash:** *url* | **ftp:** *url* | **http:** *url* | **https:** *url* | **rcp:** *url* | **scp:** *url* | **tftp:** *url* | **timeout** *seconds* | **usbflash0:** *url* | **write-delay** *seconds* }
**no ip dhcp snooping database** [ **timeout** | **write-delay** ]
abor

## Syntax Description

| | |
|---|---|
| **crashinfo:***url* | Specifies the database URL for storing entries using crashinfo. |
| **flash:***url* | Specifies the database URL for storing entries using flash. |
| **ftp:***url* | Specifies the database URL for storing entries using FTP. |
| **http:***url* | Specifies the database URL for storing entries using HTTP. |
| **https:***url* | Specifies the database URL for storing entries using secure HTTP (https). |
| **rcp:***url* | Specifies the database URL for storing entries using remote copy (rcp). |
| **scp:***url* | Specifies the database URL for storing entries using Secure Copy (SCP).<br>**Note** Before you configure SCP, you need to set the line console 0 transport output to *ssh* or *all*. |
| **tftp:***url* | Specifies the database URL for storing entries using TFTP. |
| **timeout** *seconds* | Specifies the cancel timeout interval; valid values are from 0 to 86400 seconds. |
| **usbflash0:***url* | Specifies the database URL for storing entries using USB flash. |

| | |
|---|---|
| **write-delay** *seconds* | Specifies the amount of time before writing the DHCP-snooping entries to an external server after a change is seen in the local DHCP-snooping database; valid values are from 15 to 86400 seconds. |

**Command Default**

The DHCP-snooping database is not configured.

**Command Modes**

Global configuration (config)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

**Usage Guidelines**

You must enable DHCP snooping on the interface before entering this command. Use the **ip dhcp snooping** command to enable DHCP snooping.

This example shows how to specify the database URL using TFTP:

```
Device> enable
Device# configure terminal
Device(config)#  ip dhcp snooping database tftp://10.90.90.90/snooping-rp2
Device(config)# exit
```

This example shows how to specify the amount of time before writing DHCP snooping entries to an external server:

```
evice> enable
Device# configure terminal
Device(config)#  ip dhcp snooping database write-delay 15
Device(config)# exit
```

# ip dhcp snooping information option format remote-id

To configure the option-82 remote-ID suboption, use the **ip dhcp snooping information option format remote-id** command in global configuration mode on the device to configure the option-82 remote-ID suboption. To configure the default remote-ID suboption, use the **no** form of this command.

**ip dhcp snooping information option format remote-id** {**hostname** | **string** *string*}
**no ip dhcp snooping information option format remote-id** {**hostname** | **string** *string*}

## Syntax Description

| | |
|---|---|
| **hostname** | Specify the device hostname as the remote ID. |
| **string** *string* | Specify a remote ID, using from 1 to 63 ASCII characters (no spaces). |

## Command Default

The device MAC address is the remote ID.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

You must globally enable DHCP snooping by using the **ip dhcp snooping** global configuration command for any DHCP snooping configuration to take effect.

When the option-82 feature is enabled, the default remote-ID suboption is the device MAC address. This command allows you to configure either the device hostname or a string of up to 63 ASCII characters (but no spaces) to be the remote ID.

**Note** If the hostname exceeds 63 characters, it will be truncated to 63 characters in the remote-ID configuration.

This example shows how to configure the option- 82 remote-ID suboption:

```
Device> enable
Device# configure terminal
Device(config)# ip dhcp snooping information option format remote-id hostname
Device(config)# exit
```

# ip dhcp snooping verify no-relay-agent-address

To disable the DHCP snooping feature from verifying that the relay agent address (giaddr) in a DHCP client message matches the client hardware address on an untrusted port, use the **ip dhcp snooping verify no-relay-agent-address** command in global configuration mode. To enable verification, use the **no** form of this command.

**ip dhcp snooping verify no-relay-agent-address**
**no ip dhcp snooping verify no-relay-agent-address**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The DHCP snooping feature verifies that the relay-agent IP address (giaddr) field in DHCP client message on an untrusted port is 0.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

By default, the DHCP snooping feature verifies that the relay-agent IP address (giaddr) field in DHCP client message on an untrusted port is 0; the message is dropped if the giaddr field is not 0. Use the **ip dhcp snooping verify no-relay-agent-address** command to disable the verification. Use the **no ip dhcp snooping verify no-relay-agent-address** to reenable verification.

This example shows how to enable verification of the giaddr in a DHCP client message:

```
Device> enable
Device# configure terminal
Device(config)# no ip dhcp snooping verify no-relay-agent-address
Device(config)# exit
```

# ip http access-class

To specify the access list that should be used to restrict access to the HTTP server, use the **ip http access-class** command in global configuration mode. To remove a previously configured access list association, use the **no** form of this command.

**ip http access-class** { *access-list-number* | **ipv4** { *access-list-number* | *access-list-name* } | **ipv6** *access-list-name* }
**no ip http access-class** { *access-list-number* | **ipv4** { *access-list-number* | *access-list-name* } | **ipv6** *access-list-name* }

## Syntax Description

| | |
|---|---|
| *access-list-number* | Standard IP access list number in the range 0 to 99, as configured by the **access-list** global configuration command. |
| **ipv4** | Specifies the IPv4 access list to restrict access to the secure HTTP server. |
| *access-list-name* | Name of a standard IPv4 access list, as configured by the **ip access-list** command. |
| **ipv6** | Specifies the IPv6 access list to restrict access to the secure HTTP server. |

## Command Default

No access list is applied to the HTTP server.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

If this command is configured, the specified access list is assigned to the HTTP server. Before the HTTP server accepts a connection, it checks the access list. If the check fails, the HTTP server does not accept the request for a connection.

## Examples

The following example shows how to define an access list as 20 and assign it to the HTTP server:

```
Device> enable
Device(config)# ip access-list standard 20
Device(config-std-nacl)# permit 209.165.202.130 0.0.0.255
Device(config-std-nacl)# permit 209.165.201.1 0.0.255.255
Device(config-std-nacl)# permit 209.165.200.225 0.255.255.255
Device(config-std-nacl)# exit
Device(config)# ip http access-class 20
Device(config-std-nacl)# exit
```

The following example shows how to define an IPv4 named access list as and assign it to the HTTP server.

```
Device> enable
Device(config)# ip access-list standard Internet_filter
Device(config-std-nacl)# permit 1.2.3.4
Device(config-std-nacl)# exit
```

```
Device(config)# ip http access-class ipv4 Internet_filter
Device(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **ip access-list** | Assigns an ID to an access list and enters access list configuration mode. |
| **ip http server** | Enables the HTTP 1.1 server, including the Cisco web browser user interface. |

# ip radius source-interface

To force RADIUS to use the IP address of a specified interface for all outgoing RADIUS packets, use the **ip radius source-interface** command in global configuration mode. To prevent RADIUS from using the IP address of a specified interface for all outgoing RADIUS packets, use the no form of this command.

**ip radius source-interface** *interface-name* [**vrf** *vrf-name*]
**no ip radius source-interface**

## Syntax Description

| | |
|---|---|
| *interface-name* | Name of the interface that RADIUS uses for all of its outgoing packets. |
| **vrf** *vrf-name* | (Optional) Per virtual route forwarding (VRF) configuration. |

## Command Default

No default behavior or values.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use this command to set the IP address of an interface to be used as the source address for all outgoing RADIUS packets. The IP address is used as long as the interface is in the *up* state. The RADIUS server can use one IP address entry for every network access client instead of maintaining a list of IP addresses. Radius uses the IP address of the interface that it is associated to, regardless of whether the interface is in the *up* or *down* state.

The **ip radius source-interface** command is especially useful in cases where the router has many interfaces and you want to ensure that all RADIUS packets from a particular router have the same IP address.

The specified interface should have a valid IP address and should be in the *up* state for a valid configuration. If the specified interface does not have a valid IP address or is in the *down* state, RADIUS selects a local IP that corresponds to the best possible route to the AAA server. To avoid this, add a valid IP address to the interface or bring the interface to the *up* state.

Use the **vrf** *vrf-name* keyword and argument to configure this command per VRF, which allows multiple disjoined routing or forwarding tables, where the routes of one user have no correlation with the routes of another user.

## Examples

The following example shows how to configure RADIUS to use the IP address of interface s2 for all outgoing RADIUS packets:

```
ip radius source-interface s2
```

The following example shows how to configure RADIUS to use the IP address of interface Ethernet0 for VRF definition:

```
ip radius source-interface Ethernet0 vrf vrf1
```

# ip source binding

To add a static IP source binding entry, use the **ip source binding** command. Use the **no** form of this command to delete a static IP source binding entry

**ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id*
**no ip source binding** *mac-address* **vlan** *vlan-id ip-address* **interface** *interface-id*

## Syntax Description

| | |
|---|---|
| *mac-address* | Binding MAC address. |
| **vlan** *vlan-id* | Specifies the Layer 2 VLAN identification; valid values are from 1 to 4094. |
| *ip-address* | Binding IP address. |
| **interface** *interface-id* | ID of the physical interface. |

## Command Default

No IP source bindings are configured.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

You can use this command to add a static IP source binding entry only.

The **no** format deletes the corresponding IP source binding entry. It requires the exact match of all required parameter in order for the deletion to be successful. Note that each static IP binding entry is keyed by a MAC address and a VLAN number. If the command contains the existing MAC address and VLAN number, the existing binding entry is updated with the new parameters instead of creating a separate binding entry.

This example shows how to add a static IP source binding entry:

```
Device> enable
Device# configure terminal
Device(config) ip source binding 0100.0230.0002 vlan 11 10.0.0.4 interface
gigabitethernet1/0/1
Device(config)# exit
```

# ip ssh pubkey-chain

To configure SSH Rivest, Shamir, and Adleman (SSH-RSA) keys for user and server authentication on the SSH server, use the **ip ssh pubkey-chain** command in global configuration mode. To remove SSH-RSA keys for user and server authentication on the SSH server, use the **no** form of this command.

**ip ssh pubkey-chain**
**no ip ssh pubkey-chain**

## Syntax Description

This command has no arguments or keywords.

## Command Default

SSH-RSA keys are not configured.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **ip ssh pubkey-chain**command to ensure SSH server and user public key authentication.

## Examples

The following example shows how to generate a public key:

```
Device> enable
Device# configure terminal
Device(config)# ip ssh pubkey-chain
Device(conf-ssh-pubkey)#
```

## Related Commands

| Command | Description |
|---|---|
| **ip ssh stricthostkeycheck** | Enables strict host key checking on the SSH server. |

# ip ssh source-interface

To specify the IP address of an interface as the source address for a Secure Shell (SSH) client device, use the **ip ssh source-interface** command in global configuration mode. To remove the IP address as the source address, use the **no** form of this command.

**ip ssh source-interface** *interface*
**no ip ssh source-interface** *interface*

## Syntax Description

| | |
|---|---|
| *interface* | The interface whose address is used as the source address for the SSH client. |

## Command Default

The address of the closest interface to the destination is used as the source address (the closest interface is the output interface through which the SSH packet is sent).

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Gibraltar 16.10.1 | This command was introduced. |
| Cisco IOS XE Gibraltar 16.11.1 | |

## Usage Guidelines

By specifying this command, you can force the SSH client to use the IP address of the source interface as the source address.

## Examples

In the following example, the IP address assigned to GigabitEthernet interface 1/0/1 is used as the source address for the SSH client:

```
Device> enable
Device# configure terminal
Device(config)# ip ssh source-interface GigabitEthernet 1/0/1
Device(config)# exit
```

# ip verify source

To enable IP source guard on an interface, use the **ip verify source** command in interface configuration mode. To disable IP source guard, use the **no** form of this command.

**ip verify source** [**mac-check**] [**tracking**]
**no ip verify source**

| | |
|---|---|
| **mac-check** | (Optional) Enables IP source guard with MAC address verification. |
| **tracking** | (Optional) Enables IP port security to learn static IP address learning on a port. |

## Command Default

IP source guard is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

To enable IP source guard with source IP address filtering, use the **ip verify source** interface configuration command.

To enable IP source guard with source IP address filtering and MAC address verification, use the **ip verify source mac-check** interface configuration command.

## Examples

This example shows how to enable IP source guard with source IP address filtering on an interface:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip verify source
Device(config-if)# end
```

This example shows how to enable IP source guard with MAC address verification:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip verify source mac-check
Device(config-if)# end
```

You can verify your settings by entering the **show ip verify source** command.

# ipv6 access-list

To define an IPv6 access list and to place the device in IPv6 access list configuration mode, use the **ipv6 access-list** command in global configuration mode. To remove the access list, use the **no** form of this command.

**ipv6 access-list** { *access-list-name* | **fqdn** *access-list-name* | **match-local-traffic** | **log-update threshold** *threshold-in-msgs* | **role-based** *access-list-name* }
**no ipv6 access-list** { *access-list-name* | **fqdn** *access-list-name* | **match-local-traffic** | **log-update threshold** *threshold-in-msgs* | **role-based** *access-list-name* }

## Syntax Description

| | |
|---|---|
| *access-list-name* | Name of the IPv6 access list. Names cannot contain a space or quotation mark, and must begin with an alphabetic character. The allowed length is 64 characters. |
| **fqdn** *access-list-name* | Specifies an FQDN IPv6 access list. |
| **match-local-traffic** | Enables matching for locally-generated traffic. |
| **log-update threshold** *threshold-in-msgs* | Determines how syslog messages are generated after the initial packet match.<br>• *threshold-in-msgs*: Number of packets generated. |
| **role-based** *access-list-name* | Creates a role-based IPv6 ACL. |

## Command Default

No IPv6 access list is defined.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Bengaluru 17.6.1 | This command was modified. The **fqdn** keyword was introduced. |

## Usage Guidelines

From IPv6 access list configuration mode, permit and deny conditions can be set for the defined IPv6 ACL.

**Note** IPv6 ACLs are defined by a unique name (IPv6 does not support numbered ACLs). An IPv4 ACL and an IPv6 ACL cannot share the same name.

IPv6 is automatically configured as the protocol type in **permit any any** and **deny any any** statements that are translated from global configuration mode to IPv6 access list configuration mode.

Every IPv6 ACL has implicit **permit icmp any any nd-na**, **permit icmp any any nd-ns**, and **deny ipv6 any any** statements as its last match conditions. (The first two match conditions allow for ICMPv6 neighbor discovery.) An IPv6 ACL must contain at least one entry for the implicit **deny ipv6 any any** statement to take effect. The IPv6 neighbor discovery process makes use of the IPv6 network layer service. Therefore, by default, IPv6 ACLs implicitly 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, makes use of a separate data link layer protocol. Therefore, by default, IPv4 ACLs implicitly allow ARP packets to be sent and received on an interface.

Use the **ipv6 traffic-filter** interface configuration command with the *access-list-name* argument to apply an IPv6 ACL to an IPv6 interface. Use the **ipv6 access-class** line configuration command with the *access-list-name* argument to apply an IPv6 ACL to incoming and outgoing IPv6 virtual terminal connections to and from the device.

An IPv6 ACL applied to an interface with the **ipv6 traffic-filter** command filters traffic that is forwarded—not originated from—by the device.

## Examples

The following example shows how to configure an IPv6 ACL list named list1, and place the device in IPv6 access list configuration mode:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 access-list list1
Device(config-ipv6-acl)# end
```

The following example shows how to configure an IPv6 ACL named list2 and applies the ACL to outbound traffic on Ethernet interface 0. Specifically, the first ACL entry keeps all the packets from the network FEC0:0:0:2::/64 (packets that have the site-local prefix FEC0:0:0:2 as the first 64 bits of their source IPv6 address) from exiting from Gigabit Ethernet interface 0/1/2. The second entry in the ACL permits all other traffic to exit from Ethernet interface 0. The second entry is necessary because an implicit deny all condition is at the end of each IPv6 ACL.

```
Device> enable
Device# configure terminal
Device(config)# ipv6 access-list list2 deny FEC0:0:0:2::/64 any
Device(config)# ipv6 access-list list2 permit any any
Device(config)# interface gigabitethernet 0/1/2
Device(config-if)# ipv6 traffic-filter list2 out
Device(config-if)# end
```

The following example shows how to configure an IPv6 FQDN access list using a name, along with ACL entries and domain names:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 access-list fqdn facl
Device(config-ipv6-fqdn-acl)# sequence 10 permit ipv6 host 2001:DB8::1 host dynamic
www.example1.com
Device(config-ipv6-fqdn-acl)# sequence 20 permit tcp 2001:2:2::2/64 eq ftp host dynamic
www.example2.com log
Device(config-ipv6-fqdn-acl)# sequence 30 permit udp host dynamic www.example3.com any
sequence
Device(config-ipv6-fqdn-acl)# sequence 40 deny tcp any any eq www
Device(config-ipv6-fqdn-acl)# end
```

# ipv6 snooping policy

To configure an IPv6 snooping policy and enter IPv6 snooping configuration mode, use the **ipv6 snooping policy** command in global configuration mode. To delete an IPv6 snooping policy, use the **no** form of this command.

**ipv6 snooping policy** *snooping-policy*
**no ipv6 snooping policy** *snooping-policy*

## Syntax Description

| | |
|---|---|
| *snooping-policy* | User-defined name of the snooping policy. The policy name can be a symbolic string (such as Engineering) or an integer (such as 0). |

## Command Default

An IPv6 snooping policy is not configured.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **ipv6 snooping policy** command to create an IPv6 snooping policy. When the **ipv6 snooping policy** command is enabled, the configuration mode changes to IPv6 snooping configuration mode. In this mode, the administrator can configure the following IPv6 first-hop security commands:

- The **device-role** command specifies the role of the device attached to the port.
- The **limit address-count** *maximum* command limits the number of IPv6 addresses allowed to be used on the port.
- The **protocol** command specifies that addresses should be gleaned with Dynamic Host Configuration Protocol (DHCP) or Neighbor Discovery Protocol (NDP).
- The **security-level** command specifies the level of security enforced.
- The **tracking** command overrides the default tracking policy on a port.
- The **trusted-port** command configures a port to become a trusted port; that is, limited or no verification is performed when messages are received.

This example shows how to configure an IPv6 snooping policy:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# end
```

# key chain macsec

To configure a MACsec key chain name on a device interface to fetch a Pre Shared Key (PSK), use the **key chain macsec** command in global configuration mode. To disable it, use the **no** form of this command.

**key chain** *name* **macsec**
**no key chain** *name* [**macsec** ]

## Syntax Description

| | |
|---|---|
| *name* | Name of a key chain to be used to get keys. |

## Command Default

Key chain macsec is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

This example shows how to configure MACsec key chain to fetch a 128-bit Pre Shared Key (PSK):

```
Device> enable
Device# configure terminal
Device(config)# key chain kc1 macsec
Device(config-keychain-macsec)# key 1000
Device(config-keychain-macsec)# cryptographic-algorithm aes-128-cmac
Device(config-keychain-macsec-key)# key-string fb63e0269e2768c49bab8ee9a5c2258f
Device(config-keychain-macsec-key)# end
Device#
```

This example shows how to configure MACsec key chain to fetch a 256-bit Pre Shared Key (PSK):

```
Device> enable
Device# configure terminal
Device(config)# key chain kc1 macsec
Device(config-keychain-macsec)# key 2000
Device(config-keychain-macsec)# cryptographic-algorithm aes-256-cmac
Device(config-keychain-macsec-key)# key-string c865632acb269022447c417504a1b
f5db1c296449b52627ba01f2ba2574c2878
Device(config-keychain-macsec-key)# end
Device#
```

# key config-key password-encrypt

To store a type 6 encryption key in private NVRAM, use the **key config-key password-encrypt** command in global configuration mode. To disable the encryption, use the **no** form of this command.

**key config-key password-encrypt** [*text*]
**no key config-key password-encrypt** [*text*]

## Syntax Description

*text* (Optional) **Password** or **master** key.

**Note**
We recommended that you do not use the *text* argument, and instead use interactive mode (using the **Enter** key after you enter the **key config-key password-encrypt** command) so that the preshared key is not printed anywhere and, therefore, cannot be seen.

## Command Default

Type 6 password encryption key is not stored in private NVRAM.

## Command Modes

Global configuration (config)

## Command History

| Release | Modifica |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This com introduce |

## Usage Guidelines

You can securely store plain text passwords in type 6 format in NVRAM using a CLI. Type 6 passwords are encrypted. Although the encrypted passwords can be seen or retrieved, it is difficult to decrypt them to find out the actual password. Use the **key config-key password-encrypt** command along with the **password encryption aes** command to configure and enable the password (symmetric cipher Advanced Encryption Standard [AES] is used to encrypt the keys). The password (key) configured using the **key config-key password-encrypt** command is the master encryption key that is used to encrypt all other keys in the device.

If you configure the **password encryption aes** command without configuring the **key config-key password-encrypt** command, the following message is displayed at startup or during a nonvolatile generation (NVGEN) process, such as when the **show running-config** or **copy running-config startup-config** commands are configured:

```
"Can not encrypt password. Please configure a configuration-key with 'key config-key'"
```

### Changing a Password

If the password (master key) is changed or reencrypted, use the **key config-key password-encrypt** command) for the list registry to pass the old key and the new key to the application modules that are using type 6 encryption.

### Deleting a Password

If the master key that was configured using the **key config-key password-encrypt** command is deleted from the system, a warning is displayed (and a confirm prompt is issued) stating that all type 6 passwords will become useless. As a security measure, after the passwords are encrypted, they will never be decrypted in the Cisco IOS software. However, passwords can be re-encrypted, as explained in the previous paragraph.

**Caution** If the password that is configured using the **key config-key password-encrypt** command is lost, it cannot be recovered. We, therefore, recommend that you store the password in a safe location.

### Unconfiguring Password Encryption

If you unconfigure password encryption using the **no password encryption aes** command, all the existing type 6 passwords are left unchanged, and as long as the password (master key) that was configured using the **key config-key password-encrypt** command exists, the type 6 passwords will be decrypted as and when required by the application.

### Storing Passwords

Because no one can *read* the password (configured using the **key config-key password-encrypt** command), there is no way that the password can be retrieved from the device. Existing management stations cannot *know* what it is unless the stations are enhanced to include this key somewhere, in which case, the password needs to be stored securely within the management system. If configurations are stored using TFTP, the configurations are not standalone, meaning that they cannot be loaded onto a device. Before or after the configurations are loaded onto a device, the password must be manually added (using the **key config-key password-encrypt** command). The password can be manually added to the stored configuration. However we do not recommend this because adding the password manually allows anyone to decrypt all the passwords in that configuration.

### Configuring New or Unknown Passwords

If you enter or cut and paste ciphertext that does not match the master key, or if there is no master key, the ciphertext is accepted or saved, but an alert message is displayed:

```
"ciphertext>[for username bar>] is incompatible with the configured master key."
```

If a new master key is configured, all plain keys are encrypted and made type 6 keys. The existing type 6 keys are not encrypted. The existing type 6 keys are left as is.

If the old master key is lost or is unknown, you have the option of deleting the master key using the **no key config-key password-encrypt** command. Deleting the master key causes the existing encrypted passwords to remain encrypted in the device configuration. The passwords cannot be decrypted.

## Examples

The following example shows how a type 6 encryption key is stored in NVRAM:

```
Device> enable
Device# configure terminal
Device (config)# key config-key password-encrypt
```

## Related Commands

| Command | Description |
|---|---|
| **password encryption aes** | Enables a type 6 encrypted p |

# key-hash

To specify the SSH-Rivest, Shamir, and Adleman (RSA) key type and name, use the **key-hash**command in SSH public key configuration mode. To remove the SSH-RSA public key, use the **no** form of this command.

**key-hash** *key-type key-name*
**no key-hash** [*key-type key-name*]

## Syntax Description

| | |
|---|---|
| *key-type key-name* | The SSH RSA public key type and name. |

## Command Default

SSH key type and name are not specified.

## Command Modes

SSH public key configuration (conf-ssh-pubkey-user)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced on Cisco Catalyst 9300 Series Switches. |
| Cisco IOS XE 17.16.1 | The command was updated. Support for SSH Ed25519 public-key for encryption was added. |

## Usage Guidelines

The key type must be **ssh-rsa** for configuration of private-public key pairs. You can use a hashing software to compute the hash of the public key string or you can copy the hash value from another Cisco device. Using the **key-string** command is the preferred method for entering the public key data for the first time.

This sample RPC that shows the SSH Ed25519 type key as the encryption type.

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
     message-id="urn:uuid:a75d12cd-7aa7-4296-be30-653c57367085">
  <nc:edit-config>
    <nc:target>
      <nc:running/>
    </nc:target>
    <nc:config>
      <native xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-native">
        <ip>
          <ssh nc:operation="replace">
            <pubkey-chain>
              <username>
                <name>test1</name>
                <key-hash>
                  <key-type>ssh-ed25519</key-type>
                  <key-hash-value>DFEAF10390E560AEA745CCBA53E044DE</key-hash-value>
                </key-hash>
              </username>
            </pubkey-chain>
          </ssh>
        </ip>
      </native>
    </nc:config>
  </nc:edit-config>
</nc:rpc>

Response:
```

```
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
  xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
  message-id="urn:uuid:a75d12cd-7aa7-4296-be30-653c57367085">
  <ok/>
</rpc-reply>
```

## Examples

The following example shows how to specify the SSH key type and name:

```
Device> enable
Device# configure terminal
Device(config)# ip ssh pubkey-chain
Device(conf-ssh-pubkey)# username test
Device(conf-ssh-pubkey-user)# key-hash ssh-rsa key1
Device(conf-ssh-pubkey-user))# end
Device#
```

The following example shows how to specify an Ed25519 SSH key type and name:

```
Device> enable
Device# configure terminal
Device(config)# ip ssh pubkey-chain
Device(conf-ssh-pubkey)# username user1
Device(conf-ssh-pubkey-user)# key-hash ssh-ed25519 DFEAF10390E560AEA745CCBA53E044ED
Device(conf-ssh-pubkey-user))# end
Device#
```

## Related Commands

| Command | Description |
|---|---|
| **ip ssh pubkey-chain** | Configures SSH-RSA keys for user and server authentication on the SSH server. |
| **key-string** | Specifies the SSH RSA public key of the remote peer. |
| **username** | Establishes an username-based authentication system |

# key-server

To configure MKA key-server options, use the **key-server** command in MKA-policy configuration mode. To disable MKA key-server options, use the **no** form of this command.

**key-server priority** *value*
**no key-server priority**

## Syntax Description

| | |
|---|---|
| **priority** *value* | Specifies the priority value of the MKA key-server. |

## Command Default

MKA key-server is disabled.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Examples

The following example shows how to configure the MKA key-server:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# key-server priority 33
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **confidentiality-offset** | Sets the confidentiality offset for MACsec operations. |
| **delay-protection** | Configures MKA to use delay protection in sending MKPDU. |
| **include-icv-indicator** | Includes ICV indicator in MKPDU. |
| **macsec-cipher-suite** | Configures cipher suite for deriving SAK) |
| **sak-rekey** | Configures the SAK rekey interval. |
| **send-secure-announcements** | Configures MKA to send secure announcements in sending MKPDUs. |
| **ssci-based-on-sci** | Computes SSCI based on the SCI. |
| **use-updated-eth-header** | Uses the updated Ethernet header for ICV calculation. |

# limit address-count

To limit the number of IPv6 addresses allowed to be used on the port, use the **limit address-count** command in Neighbor Discovery Protocol (NDP) inspection policy configuration mode or IPv6 snooping configuration mode. To return to the default, use the **no** form of this command.

**limit address-count** *maximum*
**no limit address-count**

## Syntax Description

| | |
|---|---|
| *maximum* | The number of addresses allowed on the port. The range is from 1 to 10000. |

## Command Default

The default is no limit.

## Command Modes

IPv6 snooping configuration (config-ipv6-snooping)

ND inspection policy configuration (config-nd-inspection)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **limit address-count** command limits the number of IPv6 addresses allowed to be used on the port on which the policy is applied. Limiting the number of IPv6 addresses on a port helps limit the binding table size. The range is from 1 to 10000.

This example shows how to define an NDP policy name as policy1, and limit the number of IPv6 addresses allowed on the port to 25:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 nd inspection policy policy1
Device(config-nd-inspection)# limit address-count 25
Device(config-nd-inspection)# end
```

This example shows how to define an IPv6 snooping policy name as policy1, and limit the number of IPv6 addresses allowed on the port to 25:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# limit address-count 25
Device(config-ipv6-snooping)# end
```

# local-domain (Parameter Map)

To configure a local domain for the Cisco Umbrella Integration feature, use the **local-domain** command in parameter-map type inspect configuration mode. To remove a local domain, use the **no** form of this command.

**local-domain** *regex_param_map_name*
**no local-domain** *regex_param_map_name*

## Syntax Description

| | |
|---|---|
| *regex_param_map_name* | Name of the regular expression parameter map. |

## Command Default

No local domain is created for the parameter map.

## Command Modes

Parameter-map type inspect configuration (config-profile)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Usage Guidelines

A maximum of 64 local domains can be configured, and the allowed domain name length is 100 characters.

## Examples

The following example shows how to configure a local domain:

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type umbrella global
Device(config-profile)# local-domain dns_bypass
```

## Related Commands

| Command | Description |
|---|---|
| **parameter-map type umbrella global** | Configures a parameter-map type in umbrella mode. |

# mab logging verbose

To filter detailed information from MAC authentication bypass (MAB) system messages, use the **mab logging verbose** command in global configuration mode. Use the no form of this command to disable logging MAB system messages.

**mab logging verbose**
**no mab logging verbose**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Detailed logging of system messages is not enabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

This command filters details, such as anticipated success, from MAC authentication bypass (MAB) system messages. Failure messages are not filtered.

To filter verbose MAB system messages:

```
Device> enable
Device# configure terminal
Device(config)# mab logging verbose
Device(config)# exit
```

You can verify your settings by entering the **show running-config** command.

## Related Commands

| Command | Description |
|---|---|
| **authentication logging verbose** | Filters details from authentication system messages. |
| **dot1x logging verbose** | Filters details from 802.1x system messages. |
| **mab logging verbose** | Filters details from MAC authentication bypass (MAB) system me |

# mab request format attribute 32

To enable VLAN ID-based MAC authentication on a device, use the **mab request format attribute 32 vlan access-vlan** command in global configuration mode. To return to the default setting, use the **no** form of this command.

**mab request format attribute 32 vlan access-vlan**
**no mab request format attribute 32 vlan access-vlan**

## Syntax Description

This command has no arguments or keywords

## Command Default

VLAN-ID based MAC authentication is disabled.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use this command to allow a RADIUS server to authenticate a new user based on the host MAC address and VLAN. Use this feature on networks with the Microsoft IAS RADIUS server. The Cisco ACS ignores this command.

This example shows how to enable VLAN-ID based MAC authentication on a device:

```
Device> enable
Device# configure terminal
Device(config)# mab request format attribute 32 vlan access-vlan
Device(config)# exit
```

## Related Commands

| Command | Description |
|---|---|
| **authentication event** | Sets the action for specific authentication events. |
| **authentication fallback** | Configures a port to use web authentication as a fallback method that do not support IEEE 802.1x authentication. |
| **authentication host-mode** | Sets the authorization manager mode on a port. |
| **authentication open** | Enables or disables open access on a port. |
| **authentication order** | Sets the order of authentication methods used on a port. |
| **authentication periodic** | Enables or disables reauthentication on a port. |
| **authentication port-control** | Enables manual control of the port authorization state. |
| **authentication priority** | Adds an authentication method to the port-priority list. |
| **authentication timer** | Configures the timeout and reauthentication parameters for an 802.1x-enabled port. |

| Command | Description |
|---|---|
| **authentication violation** | Configures the violation modes that occur when a new device a port or when a new device connects to a port with the maxi of devices already connected to that port. |
| **mab** | Enables MAC-based authentication on a port. |
| **mab eap** | Configures a port to use the Extensible Authentication Protoc |
| **show authentication** | Displays information about authentication manager events on |

# macsec-cipher-suite

To configure cipher suite for deriving Security Association Key (SAK), use the **macsec-cipher-suite** command in MKA-policy configuration mode. To disable cipher suite for SAK, use the **no** form of this command.

**macsec-cipher-suite** {**gcm-aes-128** | **gcm-aes-256** | **gcm-aes-xpn-128** | **gcm-aes-xpn-256**}
**no macsec-cipher-suite** {**gcm-aes-128** | **gcm-aes-256** | **gcm-aes-xpn-128** | **gcm-aes-xpn-256**}

## Syntax Description

| | |
|---|---|
| **gcm-aes-128** | Configures cipher suite for deriving SAK with 128-bit encryption. |
| **gcm-aes-256** | Configures cipher suite for deriving SAK with 256-bit encryption. |
| **gcm-aes-xpn-128** | Configures cipher suite for deriving SAK with 128-bit encryption for Extended Packet Numbering (XPN). |
| **gcm-aes-xpn-256** | Configures cipher suite for deriving SAK with 256-bit encryption for XPN. |

## Command Default

GCM-AES-128 encryption is enabled.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

If the device supports both GCM-AES-128 and GCM-AES-256 ciphers, it is highly recommended to define and use a user-defined MKA policy to include both or only 256 bits cipher, based on your requirements..

## Examples

The following example shows how to configure MACsec cipher suite for deriving SAK with 256-bit encryption:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# macsec-cipher-suite gcm-aes-256
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **confidentiality-offset** | Sets the confidentiality offset for MACsec operations. |
| **delay-protection** | Configures MKA to use delay protection in sending MKPDU. |
| **include-icv-indicator** | Includes ICV indicator in MKPDU. |
| **key-server** | Configures MKA key-server options. |
| **sak-rekey** | Configures the SAK rekey interval. |

| Command | Description |
|---|---|
| **send-secure-announcements** | Configures MKA to send secure announcements in sending MKPDUs. |
| **ssci-based-on-sci** | Computes SSCI based on the SCI. |
| **use-updated-eth-header** | Uses the updated Ethernet header for ICV calculation. |

# macsec network-link

To enable MACsec Key Agreement protocol (MKA) configuration on the uplink interfaces, use the **macsec network-link** command in interface configuration mode. To disable it, use the **no** form of this command.

**macsec network-link**

**no macsec network-link**

## Syntax Description

| | |
|---|---|
| **macsec network-link** | Enables MKA MACsec configuration on device interfaces using EAP-TLS authentication protocol. |

## Command Default

MACsec network-link is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

This example shows how to configure MACsec MKA on an interface using the EAP-TLS authentication protocol:

```
Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 1/0/20
Device(config-if)# macsec network-link
Device(config-if)# end
Device#
```

# match (access-map configuration)

To set the VLAN map to match packets against one or more access lists, use the **match** command in access-map configuration mode. To remove the match parameters, use the **no** form of this command.

**match** {**ip address** {*namenumber*} [*namenumber*] [*namenumber*]**. . .** | **ipv6 address** {*namenumber*} [*namenumber*] [*namenumber*]**. . .** | **mac address** {*name*} [*name*] [*name*]**. . .**}
**no match** {**ip address** {*namenumber*} [*namenumber*] [*namenumber*]**. . .** | **ipv6 address** {*namenumber*} [*namenumber*] [*namenumber*]**. . .** | **mac address** {*name*} [*name*] [*name*]**. . .**}

## Syntax Description

| | |
|---|---|
| **ip address** | Sets the access map to match packets against an IP address access list. |
| **ipv6 address** | Sets the access map to match packets against an IPv6 address access list. |
| **mac address** | Sets the access map to match packets against a MAC address access list. |
| *name* | Name of the access list to match packets against. |
| *number* | Number of the access list to match packets against. This option is not valid for MAC access lists. |

## Command Default

The default action is to have no match parameters applied to a VLAN map.

## Command Modes

Access-map configuration (config-access-map)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

You enter access-map configuration mode by using the **vlan access-map** global configuration command.

You must enter one access list name or number; others are optional. You can match packets against one or more access lists. Matching any of the lists counts as a match of the entry.

In access-map configuration mode, use the **match** command to define the match conditions for a VLAN map applied to a VLAN. Use the **action** command to set the action that occurs when the packet matches the conditions.

Packets are matched only against access lists of the same protocol type; IP packets are matched against IP access lists, IPv6 packets are matched against IPv6 access lists, and all other packets are matched against MAC access lists.

IP, IPv6, and MAC addresses can be specified for the same map entry.

## Examples

This example shows how to define and apply a VLAN access map vmap4 to VLANs 5 and 6 that will cause the interface to drop an IP packet if the packet matches the conditions defined in access list al2:

```
Device> enable
Device(config)# vlan access-map vmap4
Device(config-access-map)# match ip address al2
Device(config-access-map)# action drop
```

```
Device(config-access-map)# exit
Device(config)# vlan filter vmap4 vlan-list 5-6
Device(config)# exit
```

You can verify your settings by entering the **show vlan access-map** command.

# mka pre-shared-key

To configure MACsec Key Agreement (MKA) MACsec on a device interface using a Pre Shared Key (PSK), use the **mka pre-shared-key** command in interface configuration mode. To disable it, use the **no** form of this command.

**mka pre-shared-key key-chain** *key-chain-name* [ **fallback key-chain** *key-chain-name* ]
**no mka pre-shared-key key-chain** *key-chain-name* [ **fallback key-chain** *key-chain-name* ]

## Syntax Description

| | |
|---|---|
| **key-chain** | Enables MACsec MKA configuration on device interfaces using a primary PSK. |
| **fallback key-chain** | (Optional) Enables MACsec MKA configuration on device interfaces using a fallback PSK. |
| *key-chain-name* | Name of the key chain. |

## Command Default

MKA pre-shared-key is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Bengaluru 17.6.2 | The **fallback key-chain** keyword was introduced. |

## Usage Guidelines

When **fallback key-chain** is configured under an interface that is MACsec capable, both the primary and fallback key chains will be associated with the interface.

This example shows how to configure MKA MACsec on an interface using a primary PSK:

```
Device> enable
Device# configure terminal
Device(config)# interface Gigabitethernet 1/0/20
Device(config-if)# mka pre-shared-key key-chain kc1
Device(config-if)# end
Device#
```

# mka suppress syslogs sak-rekey

To suppress MACsec Key Agreement (MKA) secure association key (SAK) rekey messages during logging, use the **mka suppress syslogs sak-rekey** command in global configuration mode. To enable MKA SAK rekey message logging, use the **no** form of this command.

**mka suppres syslogs sak-rekey**
**no mka suppres syslogs sak-rekey**

This command has no arguments or keywords.

## Command Default

All MKA SAK syslog messages are displayed on the console.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Gibraltar 16.9.1 | This command was introduced. |

## Usage Guidelines

MKA SAK syslogs are continuously generated at every rekey interval, and when MKA is configured on multiple interfaces, the amount of syslog generated is too high. Use this command to suppress the MKA SAK syslogs.

### Example

The following example shows show to suppress MKA SAK syslog logging:

```
Device> enable
Device# configure terminal
Device(config)# mka suppress syslogs sak-rekey
```

# orgid (Parameter Map)

To configure the API organization ID used for authorization during device registration, use the **orgid** command in parameter-map type inspect configuration mode. To remove the organization ID, use the **no** form of this command.

**orgid** *value*
**no orgid**

## Syntax Description

| | |
|---|---|
| *value* | The API organization ID. You can obtain this from the Cisco Umbrella registration server. |

## Command Default

No organization ID is created for the parameter map.

## Command Modes

Parameter-map type inspect configuration (config-profile)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Cupertino 17.7.1 | This command was introduced. |

## Usage Guidelines

To perform API registration for the Umbrella Switch Connector, the **api-key**, **orgid** and **secret** commands must be configured one after the other. The values for these commands can be retrieved from the Cisco Umbrella registration server.

## Examples

The following example shows how to perform API registration for the Umbrella Switch Connector:

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type umbrella global
Device(config-profile)# api-key 5f22922exxxxxxxxx51174af822734
Device(config-profile)# orgid 26xxx16
Device(config-profile)# secret 0 a0d176ebxxxxxxxfbb343dfc4fd209
Device(config-profile)# end
```

## Related Commands

| Command | Description |
|---|---|
| **parameter-map type umbrella global** | Configures a parameter-map type in umbrella mode. |

# parameter-map type regex

To configure a parameter-map type with a regular expression to match a specific traffic pattern, use the **parameter-map type regex** command in global configuration mode. To delete a parameter-map type with a regular expression, use the **no** form of this command.

**parameter-map type regex** *parameter-map-name*
**no parameter-map type regex**

## Syntax Description

| | |
|---|---|
| *parameter-map-name* | Name of the parameter map. The name can have a maximum of 228 alphanumeric characters.<br>**Note**<br>We do not recommend the use of blank spaces. The system interprets the first blank space as the end of the parameter-map name unless the string is delimited by quotation marks. |

## Command Default

A regex parameter map is not configured.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Usage Guidelines

You can enter a regular expression to match text strings either as an exact string or by using metacharacters to match multiple variants of a text string. You can use a regular expression to match the content of certain application traffic. For example, you can match a uniform resource identifier (URI) string inside an HTTP packet using the **match request regex** command under an HTTP inspection class map.

Press **Ctrl-V** to ignore all of the special characters in the CLI, such as a question mark (?) or a tab. For example, press **d**[**Ctrl-V**]**g** to enter **d?g** in the configuration.

The following table lists the metacharacters that have special meanings.

***Table 6: regex Metacharacters***

| Character | Description | Notes |
|---|---|---|
| **.** | Dot | Matches any single character. For example, **d.g** matches dog, dag, dtg, and any word that contains those characters. |
| **(** *xxx* **)** | Subexpression | Segregates characters from surrounding characters so that you can use other metacharacters on the subexpression. For example, **d(o\|a)g** matches dog and dag, but **do\|ag** matches do and ag. A subexpression can also be used with repeat quantifiers to differentiate the characters meant for repetition. For example, **ab(xy){3}z** matches abxyxyxyz. |
| **\|** | Alternation | Matches either of the expressions that it separates. For example, **dog\|cat** matches dog or cat. |

| Character | Description | Notes |
|---|---|---|
| ? | Question mark | Indicates that there are 0 or 1 occurrence of the previous expression. For example, **lo?se** matches lse or lose.<br>**Note**<br>You must enter **Ctrl-V** and then a question mark. Otherwise, the Help function is invoked. |
| * | Asterisk | Indicates that there are 0, 1, or any number of occurrences of the previous expression. For example, **lo*se** matches lse, lose, loose, and so on. |
| + | Plus | Indicates that there is at least one occurrence of the previous expression. For example, **lo+se** matches lose and loose, but not lse. |
| **{** *x* **}** | Repeat quantifier | Repeats exactly *x* times. For example, **ab(xy){3}z** matches abxyxyxyz. |
| **{** *x* **,}** | Minimum repeat quantifier | Repeats at least *x* times. For example, **ab(xy){2,}z** matches abxyxyz, abxyxyxyz, and so on. |
| **[** *abc* **]** | Character class | Matches any character in the bracket. For example, **[abc]** matches a, b, or c. |
| **[^** *abc* **]** | Negated character class | Matches a single character that is not contained within brackets. For example, **[^abc]** matches any character other than a, b, or c, and **[^A-Z]** matches any single character that is not an uppercase letter. |
| **[** *a* **-** *c* ] | Character range class | Matches any character in the specified range. **[a-z]** matches any lowercase letter. You can mix characters and ranges. For example, **[abcq-z]** matches a, b, c, q, r, s, t, u, v, w, x, y, z, and so does **[a-cq-z]**.<br>**Note**<br>The dash (-) character is literal only if it is the last or first character within brackets, for example, **[abc-]** or **[-abc]**. |
| **“ ”** | Quotation marks | Preserves trailing or leading spaces in the string. For example, **"test"** preserves the leading space when it looks for a match. |
| ^ | Caret | Specifies the beginning of a line. |
| \ | Escape character | When preceding a literal character, matches the literal character. For example, **\[** matches the left square bracket. |
| *char* | Character | When the character is not a metacharacter, matches the literal character. |
| **\r** | Carriage return | Matches the carriage return 0x0d. |
| **\n** | New line | Matches the new line 0x0a. |
| **\t** | Tab | Matches the tab 0x09. |
| **\f** | Formfeed | Matches the form feed 0x0c. |

| Character | Description | Notes |
|---|---|---|
| \x *nn* | Escaped hexadecimal number | Matches an ASCII character using hexadecimal numbers (exactly two digits). |
| \ *nnn* | Escaped octal number | Matches an ASCII character as an octal number (exactly three digits). For example, the character **040** represents a space. |

## Examples

The following example shows how to configure and apply a regex parameter map to an HTTP application firewall parameter-map type whose URI matches any of the following regular expressions:

- ".*cmd.exe"
- ".*money"
- ".*shopping"

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type regex uri-regex-cm
Device(config-profile)# pattern ".*cmd.exe"
Device(config-profile)# pattern ".*money"
Device(config-profile)# pattern ".*shopping"
Device(config-profile)# exit
Device(config)# class-map type inspect http uri-check-cm
Device(config-cmap)# match request uri regex uri-regex-cm
Device(config-cmap)# exit
Device(config)# policy-map type inspect http uri-check-pm
Device(config-pmap)# class type inspect http uri-check-cm
Device(config-pmap-c)# reset
```

The following example shows how to configure a regex parameter map whose case-insensitive pattern matches multiple variants of the string hello:

```
Device# configure terminal
Device(config)# parameter-map type regex body_regex
Device(config-profile)# pattern ".*[Hh][Ee][Ll][Ll][Oo]"
Device(config-profile)# end
```

## Related Commands

| Command | Description |
|---|---|
| **class-map type inspect** | Creates a Layer 3 and Layer 4 or a Layer 7 (application-specific) inspect-type class map. |
| **class type inspect** | Specifies the traffic (class) on which an action is to be performed. |
| **match request regex** | Configures an HTTP firewall policy to permit or deny HTTP traffic on the basis of request messages whose URI or arguments (parameters) match a defined regular expression. |
| **parameter-map type** | Creates or modifies a parameter map. |
| **policy-map type inspect** | Creates a Layer 3 and Layer 4 or a Layer 7 (application-specific) inspect-type policy map. |

# parameter-map type umbrella global

To configure a parameter-map type in umbrella mode, use the **parameter-map type umbrella global** command in global configuration mode. To delete a parameter-map type in umbrella mode, use the **no** form of this command.

**parameter-map type umbrella global**
**no parameter-map type umbrella**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Umbrella mode parameter-map is not configured.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Examples

The following example shows how to set the parameter-map type to umbrella mode:

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type umbrella global
Device(config-profile)#
```

## Related Commands

| Command | Description |
|---|---|
| **parameter-map type** | Creates or modifies a parameter map. |

# password encryption aes

To enable a type 6 encrypted preshared key, use the **password encryption aes** command in global configuration mode. To disable password encryption, use the **no** form of this command.

**password encryption aes**
**no password encryption aes**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Preshared keys are not encrypted.

## Command Modes

Global configuration (config)

## Command History

| Release | Modifica |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This com introduce |

## Usage Guidelines

You can securely store plain text passwords in type 6 format in NVRAM using a CLI. Type 6 passwords are encrypted. Although the encrypted passwords can be seen or retrieved, it is difficult to decrypt them to find out the actual password. Use the **key config-key password-encrypt** command along with the **password encryption aes** command to configure and enable the password (symmetric cipher Advanced Encryption Standard [AES] is used to encrypt the keys). The password (key) that is configured using the **key config-key password-encrypt** command is the master encryption key that is used to encrypt all other keys in the router.

If you configure the **password encryption aes** command without configuring the **key config-key password-encrypt** command, the following message is displayed at startup or during a nonvolatile generation (NVGEN) process, such as when the **show running-config** or **copy running-config startup-config** commands are run:

```
"Can not encrypt password. Please configure a configuration-key with 'key config-key'"
```

### Changing a Password

If the password (master key) is changed or re-encrypted using the **key config-key password-encrypt** command), the list registry passes the old key and the new key to the application modules that are using type 6 encryption.

### Deleting a Password

If the master key that was configured using the **key config-key password-encrypt** command is deleted from the system, a warning is displayed (and a confirm prompt is issued) that states that all type 6 passwords will no longer be applicable. As a security measure, after the passwords are encrypted, they will never be decrypted in the Cisco IOS software. However, passwords can be re-encrypted as explained in the previous paragraph.

**Caution** If a password that is configured using the **key config-key password-encrypt** command is lost, it cannot be recovered. Therefore, the password should be stored in a safe location.

### Unconfiguring Password Encryption

If you unconfigure password encryption using the **no password encryption aes** command, all the existing type 6 passwords are left unchanged. As long as the password (master key) that was configured using the **key config-key password-encrypt** command exists, the type 6 passwords are decrypted as and when required by the application.

### Storing Passwords

Because no one can *read* the password (configured using the **key config-key password-encrypt** command), there is no way that the password can be retrieved from the router. Existing management stations cannot *know* what it is unless the stations are enhanced to include this key somewhere. Therefore, the password needs to be stored securely within the management system. If configurations are stored using TFTP, the configurations are not standalone, meaning that they cannot be loaded onto a router. Before or after the configurations are loaded onto a router, the password must be manually added (using the **key config-key password-encrypt** command). The password can be manually added to the stored configuration, but we do not recommend this because adding the password manually allows anyone to decrypt all the passwords in that configuration.

### Configuring New or Unknown Passwords

If you enter or cut and paste ciphertext that does not match the master key, or if there is no master key, the ciphertext is accepted or saved, but the following alert message is displayed:

```
"ciphertext>[for username bar>] is incompatible with the configured master key."
```

If a new master key is configured, all the plain keys are encrypted and converted to type 6 keys. The existing type 6 keys are not encrypted. The existing type 6 keys are left as is.

If the old master key is lost or unknown, you have the option of deleting the master key using the **no key config-key password-encrypt** command. This causes the existing encrypted passwords to remain encrypted in the router configuration. The passwords will not be decrypted.

## Examples

The following example shows how a type 6 encrypted preshared key is enabled:

```
Device> enable
Device# configure terminal
Device (config)# password encryption aes
```

## Related Commands

| Command | Description |
|---|---|
| **key config-key password-encrypt** | Stores a type 6 encryption key |

# pattern (Parameter Map)

To configure a matching pattern that specifies a list of domains, URL keywords, or URL metacharacters that must be allowed or blocked by the local URL filtering, use the **pattern** command in parameter-map type inspect configuration mode. To remove the matching pattern, use the **no** form of this command.

**pattern** *expression*
**no pattern** *expression*

## Syntax Description

| | |
|---|---|
| *expression* | Matching pattern argument that refers to a domain name, URL keyword, URL metacharacter entry, or a URL keyword, and URL-metacharacter combination. |

## Command Default

No pattern is created for the parameter map.

## Command Modes

Parameter-map type inspect configuration (config-profile)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Usage Guidelines

A matching pattern expression is configured for a parameter map created by the **parameter-map type regex** command.

In a pattern expression, the characters /, {, and } are not allowed. The question mark (?) character is not allowed because it is reserved for the CLI Help function. The asterisk (*) character is not allowed at the beginning of a pattern.

For URL pattern matching, the period (.) character is interpreted as a dot, and not as a wildcard entry that represents a single character, as is the case with regular expression pattern matching. Any character in the host or domain name can be allowed or blocked through URL filtering.

A URL keyword is a complete word that comes after the domain name and is between the forward slash (/) path delimiters. For example, in the URL http://www.example.com/hack/123.html, only **hack** is treated as a keyword. The entire keyword in the URL must match a pattern. For example, if you have configured a pattern named **hack**, the URL www.example.com/hacksite/123.html will not match the pattern. To match the URL, your pattern must have **hacksite**.

URL metacharacters allow pattern matching of single characters or ranges of characters to URLs, similar to the way a UNIX glob expression works. URL metacharacters are described in the following table.

***Table 7: URL Metacharacters for URL Pattern Matching***

| Character | Description |
|---|---|
| * | Asterisk: Matches any sequence of 0 or more characters. |
| [ *abc* ] | Character class: Matches any character within brackets. Character matching is case sensitive. For example, **[abc]** matches a, b, or c. |

| Character | Description |
|---|---|
| [ *a-c* ] | Character range class: Matches any character in a specified range. Character matching is case sensitive. For example, **[a-z]** matches any lowercase letter. You can also mix characters and ranges. For example, **[abcq-z]** matches a, b, c, q, r, s, t, u, v, w, x, y, z, and so does **[a-cq-z]**.<br>**Note**<br>The dash (-) character is matched only if it is the last or the first character within brackets, for example, **[abc-]** or **[-abc]**. |
| [ 0-9 ] | Numerical range class: Matches any number within brackets. For example, **[0-9]** matches 0, 1, 2, 3, 4, 5, 6, 7, 8, or 9. |

URL metacharacters are combined with domain names and URL keywords for pattern matching. For example, the pattern www.example[0-9][0-9].com can be used to block www.example01.com, www.example33.com, www.example99.com, and so on. You can combine a keyword and a metacharacter and create a matching pattern to block a URL. For example, you can use pattern hack* to block www.example.com/hacksite/123.html.

When you configure the **parameter-map type regex** command and then the **pattern** command, patterns that are specified in the **pattern** command are used as filters in General Packet Radio Service (GPRS) Tunneling Protocol (GTP) classes.

## Examples

The following example shows how to configure a matching pattern for a specified URL:

```
Device(config)# parameter-map type regex dns_bypass
Device(config-profile)# pattern www.example.com
```

The following example shows how to specify a case-insensitive pattern that matches multiple variants of the string hello:

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type regex body-regex
Device(config-profile)# pattern ".*[Hh][Ee][Ll][Ll][Oo]"
```

The following example shows an error message that appears on the console when an asterisk (*) character is specified at the beginning of a pattern:

```
Device(config)# parameter-map type regex gtp-map
Device(config-profile)# pattern *.gprs.com
%Invalid first char + or * in regex pattern
```

## Related Commands

| Command | Description |
|---|---|
| **parameter-map type regex** | Configures a regex parameter map that matches a specific regular expression pattern and enters parameter-map type inspect configuration mode. |

# permit (MAC access-list configuration)

To allow non-IP traffic to be forwarded if the conditions are matched, use the **permit** command in MAC access-list configuration mode. To remove a permit condition from the extended MAC access list, use the **no** form of this command.

{**permit** {**any** | **host***src-MAC-addr* | *src-MAC-addr mask*} {**any** | **host***dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask* | **aarp** | **amber** | **appletalk** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **lsap***lsap mask* | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp**] [**cos***cos*]
**nopermit** {**any** | **host** *src-MAC-addr* | *src-MAC-addr mask*} {**any** | **host** *dst-MAC-addr* | *dst-MAC-addr mask*} [*type mask* | **aarp** | **amber** | **appletalk** | **dec-spanning** | **decnet-iv** | **diagnostic** | **dsm** | **etype-6000** | **etype-8042** | **lat** | **lavc-sca** | **lsap** *lsap mask* | **mop-console** | **mop-dump** | **msdos** | **mumps** | **netbios** | **vines-echo** | **vines-ip** | **xns-idp**] [**cos***cos*]

## Syntax Description

| | |
|---|---|
| **any** | Denies any source or destination MAC address. |
| **host** *src-MAC-addr* \| *src-MAC-addr mask* | Specifies a host MAC address and optional subnet mask. defined address, non-IP traffic from that address is denie |
| **host** *dst-MAC-addr* \| *dst-MAC-addr mask* | Specifies a destination MAC address and optional subnet matches the defined address, non-IP traffic to that addres |
| *type mask* | (Optional) Specifies the EtherType number of a packet w identify the protocol of the packet.<br>• *type* is 0 to 65535, specified in hexadecimal.<br>• *mask* is a mask of don't care bits applied to the Ethe |
| **aarp** | (Optional) Specifies EtherType AppleTalk Address Reso to a network address. |
| **amber** | (Optional) Specifies EtherType DEC-Amber. |
| **appletalk** | (Optional) Specifies EtherType AppleTalk/EtherTalk. |
| **dec-spanning** | (Optional) Specifies EtherType Digital Equipment Corpo |
| **decnet-iv** | (Optional) Specifies EtherType DECnet Phase IV protoc |
| **diagnostic** | (Optional) Specifies EtherType DEC-Diagnostic. |
| **dsm** | (Optional) Specifies EtherType DEC-DSM. |
| **etype-6000** | (Optional) Specifies EtherType 0x6000. |
| **etype-8042** | (Optional) Specifies EtherType 0x8042. |
| **lat** | (Optional) Specifies EtherType DEC-LAT. |
| **lavc-sca** | (Optional) Specifies EtherType DEC-LAVC-SCA. |

| | |
|---|---|
| **lsap** *lsap-number mask* | (Optional) Specifies the LSAP number (0 to 65535) o the protocol of the packet.<br>The *mask* is a mask of don't care bits applied to the L |
| **mop-console** | (Optional) Specifies EtherType DEC-MOP Remote C |
| **mop-dump** | (Optional) Specifies EtherType DEC-MOP Dump. |
| **msdos** | (Optional) Specifies EtherType DEC-MSDOS. |
| **mumps** | (Optional) Specifies EtherType DEC-MUMPS. |
| **netbios** | (Optional) Specifies EtherType DEC- Network Basic |
| **vines-echo** | (Optional) Specifies EtherType Virtual Integrated Netw |
| **vines-ip** | (Optional) Specifies EtherType VINES IP. |
| **xns-idp** | (Optional) Specifies EtherType Xerox Network Syste |
| **cos** *cos* | (Optional) Specifies an arbitrary class of service (CoS CoS can be performed only in hardware. A warning m |

## Command Default

This command has no defaults. However, the default action for a MAC-named ACL is to deny.

## Command Modes

MAC-access list configuration

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Though visible in the command-line help strings, **appletalk** is not supported as a matching condition.

You enter MAC access-list configuration mode by using the **mac access-list extended** global configuration command.

If you use the **host** keyword, you cannot enter an address mask; if you do not use the **any** or **host** keywords, you must enter an address mask.

After an access control entry (ACE) is added to an access control list, an implied **deny-any-any** condition exists at the end of the list. That is, if there are no matches, the packets are denied. However, before the first ACE is added, the list permits all packets.

To filter IPX traffic, you use the *type mask* or **lsap** *lsap mask* keywords, depending on the type of IPX encapsulation being used. Filter criteria for IPX encapsulation types as specified in Novell terminology and Cisco IOS XE terminology are listed in the following table.

***Table 8: IPX Filtering Criteria***

| IPX Encapsulation Type | | Filter Criterion |
|---|---|---|
| **Cisco IOS Name** | **Novell Name** | |
| arpa | Ethernet II | EtherType 0x8137 |

| IPX Encapsulation Type | | Filter Criterion |
|---|---|---|
| **Cisco IOS Name** | **Novell Name** | |
| snap | Ethernet-snap | EtherType 0x8137 |
| sap | Ethernet 802.2 | LSAP 0xE0E0 |
| novell-ether | Ethernet 802.3 | LSAP 0xFFFF |

This example shows how to define the MAC-named extended access list to allow NetBIOS traffic from any source to MAC address 00c0.00a0.03fa. Traffic matching this list is allowed.

```
Device> enable
Device# configure terminal
Device(config)# mac access-list extended
Device(config-ext-macl)# permit any host 00c0.00a0.03fa netbios
Device(config-ext-macl)# end
```

This example shows how to remove the permit condition from the MAC-named extended access list:

```
Device> enable
Device# configure terminal
Device(config)# mac access-list extended
Device(config-ext-macl)# no permit any 00c0.00a0.03fa 0000.0000.0000 netbios
Device(config-ext-macl)# end
```

This example permits all packets with EtherType 0x4321:

```
Device> enable
Device# configure terminal
Device(config)# mac access-list extended
Device(config-ext-macl)# permit any any 0x4321 0
Device(config-ext-macl)# end
```

You can verify your settings by entering the **show access-lists** command.

## Related Commands

| Command | Description |
|---|---|
| **deny** | Denies from the MA non-IP traffic to be fo |
| **mac access-list extended** | Creates an access list traffic. |
| **show access-lists** | Displays access contr |

# protocol (IPv6 snooping)

s

To specify that addresses should be gleaned with Dynamic Host Configuration Protocol (DHCP) or Neighbor Discovery Protocol (NDP), or to associate the protocol with an IPv6 prefix list, use the **protocol** command in IPv6 snooping configuration mode. To disable address gleaning with DHCP or NDP, use the **no** form of the command.

**protocol** {**dhcp** | **ndp**}
**no protocol** {**dhcp** | **ndp**}

## Syntax Description

| | |
|---|---|
| **dhcp** | Specifies that addresses should be gleaned in Dynamic Host Configuration Protocol (DHCP) packets. |
| **ndp** | Specifies that addresses should be gleaned in Neighbor Discovery Protocol (NDP) packets. |

## Command Default

Snooping and recovery are attempted using both DHCP and NDP.

## Command Modes

IPv6 snooping configuration mode (config-ipv6-snooping)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

If an address does not match the prefix list associated with DHCP or NDP, then control packets will be dropped and recovery of the binding table entry will not be attempted with that protocol.

- Using the **no protocol** {**dhcp** | **ndp**} command indicates that a protocol will not be used for snooping or gleaning.
- If the **no protocol dhcp** command is used, DHCP can still be used for binding table recovery.
- Data glean can recover with DHCP and NDP, though destination guard will only recovery through DHCP.

This example shows how to define an IPv6 snooping policy name as policy1, and configure the port to use DHCP to glean addresses:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# protocol dhcp
Device(config-ipv6-snooping)# end
```

# radius server

To configure the RADIUS server parameters, including RADIUS accounting and authentication, use the **radius server** command in global configuration mode. Use the **no** form of this command to return to the default settings.

**radius server** *name*
**address {ipv4 | ipv6}** *ip{address | hostname}* **auth-port** *udp-port* **acct-port** *udp-port*
**key** *string*
**automate tester username** *name* { **idle-time** | **ignore-acct-port** | **ignore-auth-port** | **probe-on** }
| **retransmit** *value* | **timeout** *seconds*
**no radius server** *name*

## Syntax Description

| | |
|---|---|
| **address {ipv4 \| ipv6}** *ip{address \| hostname}* | Specifies the IP address of the RADIUS server. |
| **auth-port** *udp-port* | (Optional) Specifies the UDP port for the RADIUS authentication server. The range is from 0 to 65536. |
| **acct-port** *udp-port* | (Optional) Specifies the UDP port for the RADIUS accounting server. The range is from 0 to 65536. |
| **key** *string* | (Optional) Specifies the authentication and encryption key for all RADIUS communication between the device and the RADIUS daemon.<br>**Note**<br>The key is a text string that must match the encryption key used on the RADIUS server. Always configure the key as the last item in this command. Leading spaces are ignored, but spaces within and at the end of the key are used. If there are spaces in your key, do not enclose the key in quotation marks unless the quotation marks are part of the key. |
| **automate tester username** | (Optional) Enables automatic server testing of the RADIUS server status.<br>• *name*: Name of the server.<br>• **idle-time**: Specifies the idle time after which server state should be verified. The range is 1 to 35791 minutes, and the default is 60 minutes.<br>• **ignore-acct-port**: Specifies that the testing should not be performed on the accounting ports of the servers.<br>• **ignore-auth-port**: Specifies that the testing should not be performed on the authentication port of the servers.<br>• **probe-on**: Sends a packet to verify the server status. |
| **retransmit** *value* | (Optional) Specifies the number of times a RADIUS request is resent when the server is not responding or responding slowly. The range is 1 to 100. This setting overrides the **radius-server retransmit** global configuration command setting. |

| | |
|---|---|
| **timeout** *seconds* | (Optional) Specifies the time interval that the device waits for the RADIUS server to reply before sending a request again. The range is 1 to 1000. This setting overrides the **radius-server timeout** command. |

**Command Default**

- The UDP port for the RADIUS accounting server is 1646.
- The UDP port for the RADIUS authentication server is 1645.
- Automatic server testing is disabled.
- The timeout value is 60 minutes (1 hour).
- When automatic testing is enabled, testing occurs on the accounting and authentication UDP ports.
- The authentication and encryption key ( string) is not configured.

**Command Modes**

Global configuration (config)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Dublin 17.10.1 | The **probe-on** keyword was introduced. |

**Usage Guidelines**

- We recommend that you configure the UDP port for the RADIUS accounting server and the UDP port for the RADIUS authentication server to nondefault values.
- You can configure the authentication and encryption key by using the **key** *string* command in RADIUS server configuration mode. Always configure the key as the last item in this command.
- Use the **automate-tester username** *name* keyword to enable automatic server testing of the RADIUS server status and to specify the username to be used.

  Use the **probe-on** keyword to verify the status of a server by sending RADIUS packets. After this keyword is configured, a five-second dead timer is started and a RADIUS packet is sent to the external RADIUS server after five seconds. The server state is updated if there is a response from the external RADIUS server. If there is no response, the packets are sent out according to the timeout interval that is configured using the **radius-server timeout** command. This will continue for 180 seconds, and if there is still no response, a new dead timer is started, based on the configured **radius-server deadtime** command.

This example shows how to configure 1645 as the UDP port for the authentication server and 1646 as the UDP port for the accounting server, and configure a key string:

```
Device> enable
Device# configure terminal
Device(config)# radius server ISE
Device(config-radius-server)# address ipv4 10.1.1 auth-port 1645 acct-port 1646
Device(config-radius-server)# key cisco123
Device(config-radius-server)# end
```

# radius-server dscp

To configure DSCP marking for authentication and accounting on RADIUS servers, use the **radius-server** command. To disable DSCP marking for authentication and accounting on RADIUS servers, use the **no** form of the command.

**radius-server dscp** { **acct** *dscp_acct_value* | **auth** *dscp_auth_value* }

## Syntax Description

| | |
|---|---|
| **acct** *dscp_acct_value* | Configures RADIUS DSCP marking value for accounting. The valid range is from 1 to 63. The default value is 0. |
| **auth** *dscp_auth_value* | Configures RADIUS DSCP marking value for authentication. The valid range is from 1 to 63. The default value is 0. |

## Command Default

The DSCP marking on RADIUS packets is disabled by default.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Bengaluru 17.5.1 | This command was introduced. |

### Example

This example shows how to configure DSCP marking for authentication and accounting on RADIUS packets:

```
Device# configure terminal
Device(config)# radius-server dscp auth 10 acct 20
```

# radius-server dead-criteria

To force one or both of the criteria, used to mark a RADIUS server as dead, to be the indicated constant, use the **radius-server dead-criteria** command in global configuration mode. To disable the criteria that were set, use the **no** form of this command.

**radius-server dead-criteria** [**time** *seconds*] [**tries** *number-of-tries*]
**no radius-server dead-criteria** [**time** *seconds* | **tries** *number-of-tries*]

## Syntax Description

| | |
|---|---|
| **time** *seconds* | (Optional) Minimum amount of time, in seconds, that must elapse from the time that the device last received a valid packet from the RADIUS server to the time the server is marked as dead. If a packet has not been received since the device booted, and there is a timeout, the time criterion will be treated as though it has been met. You can configure the time to be from 1 through 120 seconds.<br>• If the *seconds* argument is not configured, the number of seconds will range from 10 to 60 seconds, depending on the transaction rate of the server.<br>**Note**<br>Both the time criterion and the tries criterion must be met for the server to be marked as dead. |
| **tries** *number-of-tries* | (Optional) Number of consecutive timeouts that must occur on the device before the RADIUS server is marked as dead. If the server performs both authentication and accounting, both types of packets will be included in the number. Improperly constructed packets will be counted as though they were timeouts. All transmissions, including the initial transmit and all retransmits, will be counted. You can configure the number of timeouts to be from 1 through 100.<br>• If the *number-of-tries* argument is not configured, the number of consecutive timeouts will range from 10 to 100, depending on the transaction rate of the server and the number of configured retransmissions.<br>**Note**<br>Both the time criterion and the tries criterion must be met for the server to be marked as dead. |

## Command Default

The number of seconds and number of consecutive timeouts that occur before the RADIUS server is marked as dead will vary, depending on the transaction rate of the server and the number of configured retransmissions.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

**Note** Both the time criterion and the tries criterion must be met for the server to be marked as dead.

The **no** form of this command has the following cases:

- If neither the *seconds* nor the *number-of-tries* argument is specified with the **no radius-server dead-criteria** command, both time and tries will be reset to their defaults.
- If the *seconds* argument is specified using the originally set value, the time will be reset to the default value range (10 to 60).
- If the *number-of-tries* argument is specified using the originally set value, the number of tries will be reset to the default value range (10 to 100).

## Examples

The following example shows how to configure the device so that it will be considered dead after 5 seconds and 4 tries:

```
Device> enable
Device# configure terminal
Device(config)# radius-server dead-criteria time 5 tries 4
```

The following example shows how to disable the time and number-of-tries criteria that were set for the **radius-server dead-criteria** command.

```
Device(config)# no radius-server dead-criteria
```

The following example shows how to disable the time criterion that was set for the **radius-server dead-criteria** command.

```
Device(config)# no radius-server dead-criteria time 5
```

The following example shows how to disable the number-of-tries criterion that was set for the **radius-server dead-criteria** command.

```
Device(config)# no radius-server dead-criteria tries 4
```

## Related Commands

| Command | Description |
|---|---|
| **debug aaa dead-criteria transactions** | Displays AAA dead-criteria transaction values. |
| **show aaa dead-criteria** | Displays dead-criteria information for a AAA server. |
| **show aaa server-private** | Displays the status of all private RADIUS servers. |
| **show aaa servers** | Displays information about the number of packets sent to and received from AAA servers. |

# radius-server deadtime

To improve RADIUS response time when some servers might be unavailable and to skip unavailable servers immediately, use the **radius-server deadtime** command in global configuration mode. To set deadtime to 0, use the **no** form of this command.

**radius-server deadtime** *minutes*
**no radius-server deadtime**

## Syntax Description

| | |
|---|---|
| *minutes* | Length of time, in minutes (up to a maximum of 1440 minutes or 24 hours), for which a RADIUS server is skipped over by transaction requests. |

## Command Default

Dead time is set to 0.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use this command to enable the Cisco IOS software to mark as *dead* any RADIUS servers that fail to respond to authentication requests, thus avoiding the wait for the request to time out before trying the next configured server. A RADIUS server marked as *dead* is skipped by additional requests for the specified duration (in minutes) or unless there are no servers not marked as *dead*.

**Note** If a RADIUS server that is marked as *dead* receives a directed-request, the directed- request is not omitted by the RADIUS server. The RADIUS server continues to process the directed-request because the request is directly sent to the RADIUS server.

The RADIUS server will be marked as dead if both of the following conditions are met:

1. A valid response has not been received from the RADIUS server for any outstanding transaction for at least the timeout period that is used to determine whether to retransmit to that server, and
2. At at least the requisite number of retransmits plus one (for the initial transmission) have been sent consecutively across all transactions being sent to the RADIUS server without receiving a valid response from the server within the requisite timeout.

## Examples

The following example specifies five minutes of deadtime for RADIUS servers that fail to respond to authentication requests:

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# radius-server deadtime 5
```

## Related Commands

| Command | Description |
|---|---|
| **deadtime (server-group configuration)** | Configures deadtime within the context of RADIUS server groups. |
| **radius-server host** | Specifies a RADIUS server host. |
| **radius-server retransmit** | Specifies the number of times that the Cisco IOS software searches the list of RADIUS server hosts before giving up. |
| **radius-server timeout** | Sets the interval for which a device waits for a server host to reply. |

# radius-server directed-request

To allow users to log in to a Cisco network access server (NAS) and select a RADIUS server for authentication, use the **radius-server directed-request** command in global configuration mode. To disable the directed-request function, use the **no** form of this command.

**radius-server directed-request** [**restricted**]
**no radius-server directed-request** [**restricted**]

## Syntax Description

| | |
|---|---|
| **restricted** | (Optional) Prevents the user from being sent to a secondary server if the specified server is not available. |

## Command Default

The User cannot log in to a Cisco NAS and select a RADIUS server for authentication.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **radius-server directed-request** command sends only the portion of the username before the "@" symbol to the host specified after the "@" symbol. In other words, with this command enabled, you can direct a request to any of the configured servers, and only the username is sent to the specified server.

**Note** If a private RADIUS server is used as the group server by configuring the **server-private** (RADIUS) command, then the **radius-server directed-request** command cannot be configured.

The following is the sequence of events to send a message to RADIUS servers:

- If the **radius-server directed-request** command is configured:
  - A request is sent to the directed server. If there are more servers with the same IP address, the request is sent only to the first server with same IP address.
  - If a response is not received, requests will be sent to all servers listed in the first method list.
  - If no response is received with the first method, the request is sent to all servers listed in the second method list until the end of the method list is reached.

**Note** To select the directed server, search the first server group in the method list for a server with the IP address provided in a directed request. If it is not available, the first server group with the same IP address from the global pool is considered.

- If the **radius-server directed-request restricted** command is configured for every server group in the method list, until the response is received from the directed server or the end of method list is reached, the following actions occur:
  - The first server with an IP address of the directed server will be used to send the request.
  - If a server with the same IP address is not found in the server group, then the first server in the global pool with the IP address of the directed-server will be used.

If the **radius-server directed-request** command is disabled using the **no radius-server directed-request** command, the entire string, both before and after the "@" symbol, is sent to the default RADIUS server. The router queries the list of servers, starting with the first one in the list. It sends the whole string, and accepts the first response from the server.

Use the **radius-server directed-request restricted** command to limit the user to the RADIUS server identified as part of the username.

If the user request has a server IP address, then the directed server forwards it to a specific server before forwarding it to the group. For example, if a user request such as user@10.0.0.1 is sent to the directed server, and if the IP address specified in this user request is the IP address of a server, the directed server forwards the user request to the specific server.

If a directed server is configured both on the server group and on the host server, and if the user request with the configured server name is sent to the directed server, the directed server forwards the user request to the host server before forwarding it to the server group. For example, if a user request of user@10.0.0.1 is sent to the directed server and 10.0.0.1 is the host server address, then the directed server forwards the user request to the host server before forwarding the request to the server group.

**Note** When the **no radius-server directed-request restricted** command is entered, only the restricted flag is removed, and the directed-request flag is retained. To disable the directed-request function, you must also enter the **no radius-server directed-request** command.

## Examples

The following example shows how to configure the directed-request function:

```
Device> enable
Device# configure terminal
Device(config)# radius server rad-1
Device(config-radius-server)# address ipv4 10.1.1.2
Device(config-radius-server)# key dummy123
Device(config-radius-server)# exit
Device(config)# radius-server directed-request
```

## Related Commands

| Command | Description |
|---|---|
| **aaa group server** | Groups different server hosts into distinct lists and distinct methods. |
| **aaa new-model** | Enables the AAA access control model. |
| **server-private** (**RADIUS**) | Configures the IP address of the private RADIUS server for the group server. |

# radius-server domain-stripping

To configure a network access server (NAS) to strip suffixes, or to strip both suffixes and prefixes from the username before forwarding the username to the remote RADIUS server, use the **radius-server domain-stripping** command in global configuration mode. To disable a stripping configuration, use the **no** form of this command.

**Note** The **ip vrf default** command must be configured in global configuration mode before the **radius-server domain-stripping** command is configured to ensure that the default VRF name is a NULL value until the defaulf vrf name is configured.

**radius-server domain-stripping** [ [**right-to-left**] [ **prefix-delimiter** *character* [ *character2* . . . *character7* ] ] [ **delimiter** *character* [ *character2* . . . *character7* ] ] | **strip-suffix** *suffix* ] [ **vrf** *vrf-name* ]
**no radius-server domain-stripping** [ [**right-to-left**] [ **prefix-delimiter** *character* [ *character2* . . . *character7* ] ] [ **delimiter** *character* [ *character2* . . . *character7* ] ] | **strip-suffix** *suffix* ] [ **vrf** *vrf-name* ]

## Syntax Description

| | |
|---|---|
| **right-to-left** | (Optional) Specifies that the NAS will apply the stripping configuration at the first delimiter found when parsing the full username from right to left. The default is for the NAS to apply the stripping configuration at the first delimiter found when parsing the full username from left to right. |
| **prefix-delimiter** *character* [*character2...character7*] | (Optional) Enables prefix stripping and specifies the character or characters that will be recognized as a prefix delimiter. Valid values for the *character* argument are @, /, $, %, \, #, and -. Multiple characters can be entered without intervening spaces. Up to seven characters can be defined as prefix delimiters, which is the maximum number of valid characters. If a \ is entered as the final or only value for the *character* argument, it must be entered as \\. No prefix delimiter is defined by default. |
| **delimiter** *character* [*character2...character7*] | (Optional) Specifies the character or characters that will be recognized as a suffix delimiter. Valid values for the *character* argument are @, /, $, %, \, #, and -. Multiple characters can be entered without intervening spaces. Up to seven characters can be defined as suffix delimiters, which is the maximum number of valid characters. If a \ is entered as the final or only value for the *character* argument, it must be entered as \\. The default suffix delimiter is the @ character. |
| **strip-suffix** *suffix* | (Optional) Specifies a suffix to strip from the username. |
| **vrf** *vrf-name* | (Optional) Restricts the domain stripping configuration to a Virtual Private Network (VPN) routing and forwarding (VRF) instance. The *vrf-name*argument specifies the name of a VRF. |

## Command Default

Stripping is disabled. The full username is sent to the RADIUS server.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **radius-server domain-stripping** command to configure the NAS to strip the domain from a username before forwarding the username to the RADIUS server. If the full username is user1@cisco.com, enabling the **radius-server domain-stripping** command results in the username "user1" being forwarded to the RADIUS server.

Use the **right-to-left** keyword to specify that the username should be parsed for a delimiter from right to left, rather than from left to right. This allows strings with two instances of a delimiter to strip the username at either delimiter. For example, if the username is user@cisco.com@cisco.net, the suffix could be stripped in two ways. The default direction (left to right) would result in the username "user" being forwarded to the RADIUS server. Configuring the **right-to-left** keyword would result in the username "user@cisco.com" being forwarded to the RADIUS server.

Use the **prefix-delimiter** keyword to enable prefix stripping and to specify the character or characters that will be recognized as a prefix delimiter. The first configured character that is parsed will be used as the prefix delimiter, and any characters before that delimiter will be stripped.

Use the **delimiter** keyword to specify the character or characters that will be recognized as a suffix delimiter. The first configured character that is parsed will be used as the suffix delimiter, and any characters after that delimiter will be stripped.

Use **strip-suffix** *suffix* to specify a particular suffix to strip from usernames. For example, configuring the **radius-server domain-stripping strip-suffix cisco.net** command would result in the username user@cisco.net being stripped, while the username user@cisco.com will not be stripped. You may configure multiple suffixes for stripping by issuing multiple instances of the **radius-server domain-stripping** command. The default suffix delimiter is the @ character.

**Note** Issuing the **radius-server domain-stripping s trip-suffix** *suffix* command disables the capacity to strip suffixes from all domains. Both the suffix delimiter and the suffix must match for the suffix to be stripped from the full username. The default suffix delimiter of @ will be used if you do not specify a different suffix delimiter or set of suffix delimiters using the **delimiter**keyword.

To apply a domain-stripping configuration only to a specified VRF, use the **vrf** *vrf-name* option.

The interactions between the different types of domain stripping configurations are as follows:

- You may configure only one instance of the **radius-server domain-stripping**[**right-to-left**] [**prefix-delimiter** *character* [*character2...character7*]] [**delimiter** *character* [*character2...character7*]] command.
- You may configure multiple instances of the **radius-server domain-stripping**[**right-to-left**] [**prefix-delimiter** *character* [*character2...character7*]] [**delimiter** *character* [*character2...character7*]] [**vrf** *vrf-name*] command with unique values for **vrf** *vrf-name*.
- You may configure multiple instances of the **radius-server domain-stripping strip-suffix** *suffix*[**vrf** *per-vrf*] command to specify multiple suffixes to be stripped as part of a global or per-VRF ruleset.

- Issuing any version of the **radius-server domain-stripping** command automatically enables suffix stripping using the default delimiter character @ for that ruleset, unless a different delimiter or set of delimiters is specified.
- Configuring a per-suffix stripping rule disables generic suffix stripping for that ruleset. Only suffixes that match the configured suffix or suffixes will be stripped from usernames.

## Examples

The following example configures the router to parse the username from right to left and sets the valid suffix delimiter characters as @, \, and $. If the full username is cisco/user@cisco.com$cisco.net, the username "cisco/user@cisco.com" will be forwarded to the RADIUS server because the $ character is the first valid delimiter encountered by the NAS when parsing the username from right to left.

```
radius-server domain-stripping right-to-left delimiter @\$
```

The following example configures the router to strip the domain name from usernames only for users associated with the VRF instance named abc. The default suffix delimiter @ will be used for generic suffix stripping.

```
radius-server domain-stripping vrf abc
```

The following example enables prefix stripping using the character / as the prefix delimiter. The default suffix delimiter character @ will be used for generic suffix stripping. If the full username is cisco/user@cisco.com, the username "user" will be forwarded to the RADIUS server.

```
radius-server domain-stripping prefix-delimiter /
```

The following example enables prefix stripping, specifies the character / as the prefix delimiter, and specifies the character # as the suffix delimiter. If the full username is cisco/user@cisco.com#cisco.net, the username "user@cisco.com" will be forwarded to the RADIUS server.

```
radius-server domain-stripping prefix-delimiter / delimiter #
```

The following example enables prefix stripping, configures the character / as the prefix delimiter, configures the characters $, @, and # as suffix delimiters, and configures per-suffix stripping of the suffix cisco.com. If the full username is cisco/user@cisco.com, the username "user" will be forwarded to the RADIUS server. If the full username is cisco/user@cisco.com#cisco.com, the username "user@cisco.com" will be forwarded.

```
radius-server domain-stripping prefix-delimiter / delimiter $@#
radius-server domain-stripping strip-suffix cisco.com
```

The following example configures the router to parse the username from right to left and enables suffix stripping for usernames with the suffix cisco.com. If the full username is cisco/user@cisco.net@cisco.com, the username "cisco/user@cisco.net" will be forwarded to the RADIUS server. If the full username is cisco/user@cisco.com@cisco.net, the full username will be forwarded.

```
radius-server domain-stripping right-to-left
radius-server domain-stripping strip-suffix cisco.com
```

The following example configures a set of global stripping rules that will strip the suffix cisco.com using the delimiter @, and a different set of stripping rules for usernames associated with the VRF named myvrf:

```
radius-server domain-stripping strip-suffix cisco.com
!
radius-server domain-stripping prefix-delimiter # vrf myvrf
radius-server domain-stripping strip-suffix cisco.net vrf myvrf
```

**Related Commands**

| Command | Description |
|---|---|
| **aaa new-model** | Enables the AAA access control model. |
| **ip vrf** | Defines a VRF instance and enters VRF configuration mode. |
| **tacacs-server domain-stripping** | Configures a router to strip a prefix or suffix from the username before forwarding the username to the TACACS+ server. |

# sak-rekey

To configure the Security Association Key (SAK) rekey time interval for a defined MKA policy, use the **sak-rekey** command in MKA-policy configuration mode. To stop the SAK rekey timer, use the **no** form of this command.

**sak-rekey** {**interval** *time-interval* | **on-live-peer-loss**}
**no sak-rekey** {**interval** | **on-live-peer-loss**}

## Syntax Description

| | |
|---|---|
| **interval** *time-interval* | SAK rekey interval in seconds.<br>The range is from 30 to 65535, and the default is 0. |
| **on-live-peer-loss** | Peer loss from the live membership. |

## Command Default

The SAK rekey timer is disabled. The default is 0.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Fuji 16.8.1a | This command was introduced. |

## Examples

The following example shows how to configure the SAK rekey interval:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# sak-rekey interval 300
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **confidentiality-offset** | Sets the confidentiality offset for MACsec operations. |
| **delay-protection** | Configures MKA to use delay protection in sending MKPDU. |
| **include-icv-indicator** | Includes ICV indicator in MKPDU. |
| **key-server** | Configures MKA key-server options. |
| **macsec-cipher-suite** | Configures cipher suite for deriving SAK. |
| **send-secure-announcements** | Configures MKA to send secure announcements in sending MKPDUs. |
| **ssci-based-on-sci** | Computes SSCI based on the SCI. |
| **use-updated-eth-header** | Uses the updated Ethernet header for ICV calculation. |

# secret (Parameter Map)

To configure the API secret password used for authorization during device registration, use the **secret** command in parameter-map type inspect configuration mode. To remove the secret password, use the **no** form of this command.

**secret** { **0** | **6** *unencrypted-password* }

**no secret** { **0** | **6** *unencrypted-password* }

## Syntax Description

| | |
|---|---|
| **0** | Specifies that an unencrypted password or secret (depending on the configuration) follows. |
| **6** | Specifies that an encrypted password follows. |
| *unencrypted-password* | The unencrypted (cleartext) user password. |

## Command Default

No password is created for the parameter map.

## Command Modes

Parameter-map type inspect configuration (config-profile)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Cupertino 17.7.1 | This command was introduced. |

## Usage Guidelines

To perform API registration for the Umbrella Switch Connector, the **api-key**, **orgid** and **secret** commands must be configured one after the other. The values for these commands can be retrieved from the Cisco Umbrella registration server.

## Examples

The following example shows how to perform API registration for the Umbrella Switch Connector:

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type umbrella global
Device(config-profile)# api-key 5f22922exxxxxxxxx51174af822734
Device(config-profile)# orgid 26xxx16
Device(config-profile)# secret 0 a0d176ebxxxxxxxfbb343dfc4fd209
Device(config-profile)# end
```

## Related Commands

| Command | Description |
|---|---|
| **parameter-map type umbrella global** | Configures a parameter-map type in umbrella mode. |

# security level (IPv6 snooping)

To specify the level of security enforced, use the **security-level** command in IPv6 snooping policy configuration mode.

**security level** {**glean** | **guard** | **inspect**}

## Syntax Description

| | |
|---|---|
| **glean** | Extracts addresses from the messages and installs them into the binding table without performing any verification. |
| **guard** | Performs both glean and inspect. Additionally, RA, and DHCP server messages are rejected unless they are received on a trusted port or another policy authorizes them. |
| **inspect** | Validates messages for consistency and conformance; in particular, address ownership is enforced. Invalid messages are dropped. |

## Command Default

The default security level is guard.

## Command Modes

IPv6 snooping configuration (config-ipv6-snooping)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

This example shows how to define an IPv6 snooping policy name as policy1 and configure the security level as inspect:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# security-level inspect
Device(config-ipv6-snooping)# end
```

# security passthru

To modify the IPsec pass-through, use the **security passthru** command. To disable, use the no form of the command.

**security passthru** *ip-address*
**no security passthru**

## Syntax Description

| | |
|---|---|
| *ip-address* | IP address of the IPsec gateway that is terminating the VPN tunnel. |

## Command Default

None.

## Command Modes

wlan

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

This example shows how to modify IPSec pass-through.

```
Device> enable
Device# configure terminal
Device(config)# security passthrough 10.1.1.1
```

# send-secure-announcements

To enable MKA to send secure announcements in MACsec Key Agreement Protocol Data Units (MKPDUs), use the **send-secure-announcements** command in MKA-policy configuration mode. To disable sending of secure announcements, use the **no** form of this command.

**send-secure-announcements**
**no send-secure-announcements**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Secure announcements in MKPDUs is disabled.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Fuji 16.9.1 | This command was introduced. |

## Usage Guidelines

Secure announcements revalidate the MACsec Cipher Suite capabilities which were shared previously through unsecure announcements.

## Examples

The following example shows how to enable sending of secure announcements:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# send-secure-announcements
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **confidentiality-offset** | Sets the confidentiality offset for MACsec operations. |
| **delay-protection** | Configures MKA to use delay protection in sending MKPDU. |
| **include-icv-indicator** | Includes ICV indicator in MKPDU. |
| **key-server** | Configures MKA key-server options. |
| **macsec-cipher-suite** | Configures cipher suite for deriving SAK. |
| **sak-rekey** | Configures the SAK rekey interval. |
| **ssci-based-on-sci** | Computes SSCI based on the SCI. |
| **use-updated-eth-header** | Uses the updated ethernet header for ICV calculation. |

# server-private (RADIUS)

To configure the IP address of the private RADIUS server for the group server, use the **server-private** command in RADIUS server-group configuration mode. To remove the associated private server from the authentication, authorization, and accounting (AAA) group server, use the **no** form of this command.

**server-private** *ip-address* [**auth-port** *port-number* | **acct-port** *port-number*] [**non-standard**] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]
**no server-private** *ip-address* [**auth-port** *port-number* | **acct-port** *port-number*] [**non-standard**] [**timeout** *seconds*] [**retransmit** *retries*] [**key** *string*]

## Syntax Description

| | |
|---|---|
| *ip-address* | IP address of the private RADIUS server host. |
| **auth-port** *port-number* | (Optional) User Datagram Protocol (UDP) destination port for authentication requests. The default value is 1645. |
| **acct-port** *port-number* | Optional) UDP destination port for accounting requests. The default value is 1646. |
| **non-standard** | (Optional) RADIUS server is using vendor-proprietary RADIUS attributes. |
| **timeout** *seconds* | (Optional) Time interval (in seconds) that the device waits for the RADIUS server to reply before retransmitting. This setting overrides the global value of the **radius-server timeout** command. If no timeout value is specified, the global value is used. |
| **retransmit** *retries* | (Optional) Number of times a RADIUS request is resent to a server, if that server is not responding or responding slowly. This setting overrides the global setting of the **radius-server retransmit** command. |
| **key** *string* | (Optional) Authentication and encryption key used between the device and the RADIUS daemon running on the RADIUS server. This key overrides the global setting of the **radius-server key** command. If no key string is specified, the global value is used.<br>The *string* can be **0** (specifies that an unencrypted key follows), **6** (specifies that an advanced encryption scheme [AES] encrypted key follows), **7** (specifies that a hidden key follows), or a line specifying the unencrypted (clear-text) server key. |

## Command Default

If server-private parameters are not specified, global configurations will be used; if global configurations are not specified, default values will be used.

## Command Modes

RADIUS server-group configuration (config-sg-radius)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **server-private** command to associate a particular private server with a defined server group. To prevent possible overlapping of private addresses between virtual route forwarding (VRF) instances, private

servers (servers with private addresses) can be defined within the server group and remain hidden from other groups, while the servers in the global pool (default "radius" server group) can still be referred to by IP addresses and port numbers. Thus, the list of servers in server groups includes references to the hosts in the global configuration and the definitions of private servers.

**Note**

- If the **radius-server directed-request** command is configured, then a private RADIUS server cannot be used as the group server by configuring the **server-private** (RADIUS) command.
- Creating or updating AAA server statistics record for private RADIUS servers are not supported. If private RADIUS servers are used, then error messages and tracebacks will be encountered, but these error messages or tracebacks do not have any impact on the AAA RADIUS functionality. To avoid these error messages and tracebacks, configure public RADIUS server instead of private RADIUS server.

Use the **password encryption aes** command to configure type 6 AES encrypted keys.

## Examples

The following example shows how to define the sg_water RADIUS group server and associate private servers with it:

```
Device> enable
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa group server radius sg_water
Device(config-sg-radius)# server-private 10.1.1.1 timeout 5 retransmit 3 key xyz
Device(config-sg-radius)# server-private 10.2.2.2 timeout 5 retransmit 3 key xyz
Device(config-sg-radius)# end
```

## Related Commands

| Command | Description |
|---|---|
| **aaa group server** | Groups different server hosts into distinct lists and distinct methods. |
| **aaa new-model** | Enables the AAA access control model. |
| **password encryption aes** | Enables a type 6 encrypted preshared key. |
| **radius-server host** | Specifies a RADIUS server host. |
| **radius-server directed-request** | Allows users to log in to a Cisco NAS and select a RADIUS server for authentication. |

# server-private (TACACS+)

To configure the IPv4 or IPv6 address of the private TACACS+ server for the group server, use the **server-private** command in server-group configuration mode. To remove the associated private server from the authentication, authorization, and accounting (AAA) group server, use the **no** form of this command.

**server-private** { *ipv4-address* | *ipv6-address* | **fqdn** } [ **nat** ] [ **single-connection** ] [ **port** *port-number* ] [ **timeout** *seconds* ] **key** [ **0** | **7** ] *string*
**no server-private**

## Syntax Description

| | |
|---|---|
| *ip4-address* | IPv4 address of the private TACACS+ server host. |
| *ip6-address* | IPv6 address of the private TACACS+ server host. |
| **fqdn** | Fully qualified domain name (fqdn) of the private TACACS+ server host for address resolution from the Domain Name Server (DNS) |
| **nat** | (Optional) Specifies the port Network Address Translation (NAT) address of the remote device. This address is sent to the TACACS+ server. |
| **single-connection** | (Optional) Maintains a single TCP connection between the router and the TACACS+ server. |
| **timeout** *seconds* | (Optional) Specifies a timeout value for the server response. This value overrides the global timeout value set with the **tacacs-server timeout** command for this server only. |
| **port** *port-number* | (Optional) Specifies a server port number. This option overrides the default, which is port 49. |
| **key** [**0**\|**7**] *string* | (Optional) Specifies an authentication and encryption key. This key must match the key used by the TACACS+ daemon. Specifying this key overrides the key set by the global tacacs-server key command for this server only.<br>If no number or 0 is entered, the *string* that is entered is considered to be plain text. If 7 is entered, the *string* that is entered is considered to be encrypted text. |

## Command Default

If server-private parameters are not specified, global configurations will be used; if global configurations are not specified, default values will be used.

## Command Modes

TACACS+ server-group configuration (config-sg-tacacs+)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **server-private** command to associate a particular private server with a defined server group. To prevent possible overlapping of private addresses between virtual route forwardings (VRFs), private servers (servers with private addresses) can be defined within the server group and remain hidden from other groups, while the servers in the global pool (default "TACACS+" server group) can still be referred to by IP addresses

and port numbers. Thus, the list of servers in server groups includes references to the hosts in the global configuration and the definitions of private servers.

The following example shows how to define the tacacs1 TACACS+ group server and associate private servers with it:

```
Device> enable
Device# configure terminal
Device(config)# ip tacacs source-interface GigabitEthernet1/0/23 vrf vrf17
Device(config-sg-tacacs+)# aaa group server tacacs tacacs1
Device(config-sg-tacacs+)# server-private 10.1.1.1 port 19 key cisco
Device(config-sg-tacacs+)# ip vrf forwarding vrf17
Device(config-sg-tacacs+)# ip tacacs source-interface loopback0
Device(config-sg-tacacs+)# end
```

## Related Commands

| Command | Description |
|---|---|
| **aaa group server** | Groups different server hosts into distinct lists and distinct methods. |
| **aaa new-model** | Enables the AAA access control model. |
| **ip tacacs source-interface** | Uses the IP address of a specified interface for all outgoing TACACS+ packets. |
| **ip vrf forwarding (server-group)** | Configures the VRF reference of an AAA TACACS+ server group. |

# show aaa cache group

To display all the cache entries stored by the AAA cache, use the **show aaa cache group** command in privileged EXEC mode.

**show aaa cache group** *name* { **all** | **profile** *name* }

## Syntax Description

| | |
|---|---|
| *name* | Text string representing a cache server group. |
| **all** | Displays all the server group profile details. |
| **profile** *name* | Displays the specified individual server group profile details. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **IOSD AAA Auth Cache entries** section of the command output displays Cisco IOSd-related AAA authentication cache entries that get populated when AAA authentication cache is used as the authentication method for Cisco IOSd use-cases like PPP, login, and so on. The **SMD AAA Auth Cache entries** section of the command output displays SMD AAA authentication cache entries that get populated when AAA authentication cache is being used as the authentication method for session manager daemon (SMD) use cases, such as 802.1x, MAB, and so on. The **show aaa cache group** command displays Cisco IOSd use cases-related AAA authentication cache entries first, followed by SMD use cases-related AAA authentication cache entries.

## Examples

The following example shows how to display all the cache entries for a group. The fields are self-explanatory.

```
Device# show aaa cache group radiusGroup all

IOSD AAA Auth Cache entries:
------------------------------------------------------------
Entries in Profile dB radiusGroup for exact match:
No entries found in Profile dB

SMD AAA Auth Cache entries:
------------------------------------------------------------
***Total number of AAA Auth cache entries is 3

MAC ADDR: 5C85.7E31.756C
Profile Name: CACHE-PROFILE
User Name: test
Timeout: 86400

MAC ADDR: AABB.CCDD.EE00
Profile Name: CACHE-PROFILE
User Name: cache1
Timeout: 86400
```

```
MAC ADDR: AABB.CCDD.EE01
Profile Name: CACHE-PROFILE
User Name: cache2
Timeout: 86400
```

## Related Commands

| Command | Description |
|---|---|
| **clear aaa cache group** | Clears individual entries or all the entries in the cache. |
| **debug aaa cache group** | Debugs the caching mechanism and ensures that entries are cached from AAA server responses, and found when queried. |

# show aaa clients

To display authentication, authorization, and accounting (AAA) client statistics, use the **show aaa clients** command.

**show aaa clients** [**detailed**]

## Syntax Description

| | |
|---|---|
| **detailed** | (Optional) Shows detailed AAA client statistics. |

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

This is an example of output from the **show aaa clients** command:

```
Device> enable
Device# show aaa clients

Dropped request packets: 0
```

# show aaa command handler

To display authentication, authorization, and accounting (AAA) command handler statistics, use the **show aaa command handler** command.

**show aaa command handler**

## Syntax Description

This command has no aruguments or keywords.

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

This is an example of output from the **show aaa command handler** command:

```
Device# show aaa command handler

AAA Command Handler Statistics:
    account-logon: 0, account-logoff: 0
    account-query: 0, pod: 0
    service-logon: 0, service-logoff: 0
    user-profile-push: 0, session-state-log: 0
    reauthenticate: 0, bounce-host-port: 0
    disable-host-port: 0, update-rbacl: 0
    update-sgt: 0, update-cts-policies: 0
    invalid commands: 0
    async message not sent: 0
```

# show aaa common-criteria policy

To display AAA common criteria security policy details, use the **show aaa common-criteria policy** command in privileged EXEC mode.

**show aaa common-criteria policy** { **name** *policy-name* | **all** }

## Syntax Description

| | |
|---|---|
| **name** *policy-name* | Specifies the password security details for a specific policy. |
| **all** | Specifies the password security details for all the configured policies. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **show aaa common-criteria policy** command to display the security policy details for a specific policy or for all the configured policies.

## Examples

The following is a sample output from the **show aaa common-criteria policy** command:

```
Device# show aaa common-criteria policy name policy1

Policy name: policy1
Minimum length: 1
Maximum length: 64
Upper Count: 20
Lower Count: 20
Numeric Count: 5
Special Count: 2
Number of character changes 4
Valid forever. User tied to this policy will not expire.
```

The following is a sample output from the **show aaa common-criteria policy all** command:

```
Device# show aaa common-criteria policy all
==========================================================

Policy name: policy1
Minimum length: 1
Maximum length: 64
Upper Count: 20
Lower Count: 20
Numeric Count: 5
Special Count: 2
Number of character changes 4
Valid forever. User tied to this policy will not expire.
==========================================================
Policy name: policy2
Minimum length: 1
Maximum length: 34
```

```
Upper Count: 10
Lower Count: 5
Numeric Count: 4
Special Count: 2
Number of character changes 4
Valid forever. User tied to this policy will not expire.
=========================================================
```

The following table describes the significant fields shown in the display.

***Table 9: show aaa common-criteria policy all Field Descriptions***

| Field | Description |
|---|---|
| Policy name | Name of the configured security policy. |
| Minimum length | Minimum length of the password. |
| Maximum length | Maximum length of the password. |
| Upper Count | Number of uppercase characters. |
| Lower Count | Number of lowercase characters. |
| Numeric Count | Number of numeric characters. |
| Special Count | Number of special characters. |
| Number of character changes | Number of changed characters between old and new passwords. |

## Related Commands

| Command | Description |
|---|---|
| **aaa common-criteria policy** | Configures an AAA common criteria security policy. |
| **debug aaa common-criteria** | Enables debugging for the AAA common criteria password security policies. |

# show aaa dead-criteria

To display dead-criteria detection information for an authentication, authorization, and accounting (AAA) server, use the **show aaa dead-criteria** command in privileged EXEC mode.

**show aaa dead-criteria** { **security-protocol** *ip-address* | *server-name* } [**auth-port** *port-number*] [**acct-port** *port-number*][*server-group-name*]

## Syntax Description

| | |
|---|---|
| **security-protocol** | Security protocol of the specified AAA server. Currently, the only protocol that is supported is RADIUS. |
| *ip-address* | IP address of the specified AAA server. |
| *server-name* | Name of the specified AAA server. |
| **auth-port** | (Optional) Authentication port for the RADIUS server that was specified. |
| *port-number* | (Optional) Number of the authentication port. The default is 1645 (for a RADIUS server). |
| **acct-port** | (Optional) Accounting port for the RADIUS server that was specified. |
| *port-number* | (Optional) Number of the accounting port. The default is 1646 (for a RADIUS server). |
| *server-group-name* | (Optional) Server group with which the specified server is associated. The default is *radius* (for a RADIUS server). |

## Command Default

Currently, the *port-number* argument for the **auth-port** keyword and the *port-number* argument for the **acct-port** keyword default to 1645 and 1646, respectively. The default for the *server-group-name* argument is radius.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Dublin 17.11.1 | The *server-name* option was added to the command. |

## Usage Guidelines

Multiple RADIUS servers having the same IP address can be configured on a device. The **auth-port** and **acct-port** keywords are used to differentiate the servers. The dead-detect interval of a server that is associated with a specified server group can be obtained by using the **server-group-name** keyword. (The dead-detect interval and retransmit values of a RADIUS server are set on the basis of the server group to which the server belongs. The same server can be part of multiple server groups.)

## Examples

The following example shows that dead-criteria-detection information has been requested for a RADIUS server at the IP address 192.0.2.1:

```
Device# show aaa dead-criteria radius 192.0.2.1 radius

RADIUS Server Dead Critieria:
=============================
Server Details:
    Address : 192.0.2.1
    Auth Port : 1645
    Acct Port : 1646
Server Group : radius
Dead Criteria Details:
    Configured Retransmits : 62
    Configured Timeout : 27
    Estimated Outstanding Transactions: 5
    Dead Detect Time : 25s
    Computed Retransmit Tries: 22
    Statistics Gathered Since Last Successful Transaction
=====================================================
Max Computed Outstanding Transactions: 5
Max Computed Dead Detect Time: 25s
Max Computed Retransmits : 22
```

## Examples

The following example shows that dead-criteria-detection information has been requested for a RADIUS server named ISE:

```
Device# show aaa dead-criteria radius server-name ISE

RADIUS Server Dead Criteria:
============================
Server Details:
    Address   : 192.0.2.2
    Auth Port : 1645
    Acct Port : 1646
Server Group  : radius
VRF           : Mgmt-vrf
Dead Criteria Details:
    Configured Retransmits   : 3
    Configured Timeout       : 5
    Estimated Outstanding Access Transactions: 0
    Estimated Outstanding Accounting Transactions: 0
    Dead Detect Time         : 5s
    Computed Retransmit Tries: 4
Statistics Gathered Since Last Successful Transaction
=====================================================
    Max Computed Outstanding Transactions: 1
    Max Computed Dead Detect Time: 10s
    Max Computed Retransmits : 10
```

The **Max Computed Dead Detect Time** is displayed in seconds. The other fields shown in the display are self-explanatory.

## Related Commands

| Command | Description |
|---|---|
| **debug aaa dead-criteria transactions** | Displays AAA dead-criteria transaction values. |
| **radius-server dead-criteria** | Forces one or both of the criteria used to mark a RADIUS server as dead to be the indicated the constant. |
| **show aaa server-private** | Displays the status of all private RADIUS servers. |

| Command | Description |
| --- | --- |
| **show aaa servers** | Displays information about the number of packets sent to and received from AAA servers. |

# show aaa local

To display authentication, authorization, and accounting (AAA) local method options, use the **show aaa local** command.

**show aaa local** {**netuser** {*name* | **all** } | **statistics** | **user lockout**}

## Syntax Description

| | |
|---|---|
| **netuser** | Specifies the AAA local network or guest user database. |
| *name* | Network user name. |
| **all** | Specifies the network and guest user information. |
| **statistics** | Displays statistics for local authentication. |
| **user lockout** | Specifies the AAA local locked-out user. |

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

This is an example of output from the **show aaa local statistics** command:

```
Device# show aaa local statistics

Local EAP statistics

EAP Method         Success       Fail
---------------------------------------
Unknown                  0          0
EAP-MD5                  0          0
EAP-GTC                  0          0
LEAP                     0          0
PEAP                     0          0
EAP-TLS                  0          0
EAP-MSCHAPV2             0          0
EAP-FAST                 0          0

Requests received from AAA:                 0
Responses returned from EAP:                0
Requests dropped (no EAP AVP):              0
Requests dropped (other reasons):           0
Authentication timeouts from EAP:           0

Credential request statistics
Requests sent to backend:                   0
Requests failed (unable to send):           0
Authorization results received

  Success:                                  0
```

```
Fail:                                    0
```

# show aaa servers

To display all authentication, authorization, and accounting (AAA) servers as seen by the AAA server MIB, use the **show aaa servers** command.

**show aaa servers [private** | **public** | [**detailed]]**

## Syntax Description

| | |
|---|---|
| **detailed** | (Optional) Displays private AAA servers as seen by the AAA server MIB. |
| **public** | (Optional) Displays public AAA servers as seen by the AAA server MIB. |
| **detailed** | (Optional) Displays detailed AAA server statistics. |

## Command Modes

User EXEC (>)

Privileged EXEC (>)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Fuji 16.9.1 | The output of the command was updated. |

## Examples

The following is a sample output from the **show aaa servers** command:

```
Device# show aaa servers

Bad authenticators: 0
RADSEC: Packet count since last idletimeout 0,
Send handshake count 0,
Handshake Success 0,
Total Packets Transmitted 0,
Total Packets Received 0,
Total Connection Resets 9,
Connection Reset due to idle timeout 0,
Connection Reset due to No Response 0,
Connection Reset due to Malformed packet 0,
Connection Reset by Peer 0,
```

# show aaa sessions

To display authentication, authorization, and accounting (AAA) sessions as seen by the AAA Session MIB, use the **show aaa sessions** command.

**show aaa sessions**

## Syntax Description

This command has no arguments or keywords.

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

The following is sample output from the **show aaa sessions** command:

```
Device# show aaa sessions

Total sessions since last reload: 7
Session Id: 4007
   Unique Id: 4025
   User Name: *not available*
   IP Address: 0.0.0.0
   Idle Time: 0
   CT Call Handle: 0
```

# show access-lists fqdn statistics

To display the latency statistics of the fully qualified domain name (FQDN) ACLs, use the **show access-lists fqdn statistics** command in privileged EXEC mode.

**show access-lists fqdn statistics**

## Syntax Description

This command has no arguments or keywords.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Bengaluru 17.6.1 | This command was introduced. |

## Examples

The following is a sample output from the **show access-lists fqdn statistics** command:

```
Device# show access-lists fqdn statistics

IPv4 ACL programming requests: 231
IPv4 ACL programming timeout: 24
IPv4 ACL programming latency:
Min 2 ms
Max 134 ms
Avg 56 ms
IPv6 ACL programming requests: 178
IPv6 ACL programming timeout: 6
IPv6 ACL programming latency:
Min 2 ms
Max 253 ms
Avg 24 ms
```

## Related Commands

| Command | Description |
|---|---|
| **show fqdn** | Displays the resolved IP addresses for the FQDN entries from the local cache database. |

# show access-session

To display information about Session Aware Networking sessions, use the **show access-session** command in privileged EXEC mode.

**show access-session** { **database** | **brief** | **cache** | **event-logging** [ **mac** *mac-adress* | **display-all** | **unauth** ] | **fqdn** [ **passthru-domain-list** | **list-domain** *list-domain* | **fqdn-maps** ] | **history** | **info** | **interface** *interface-name interface-number* | **mac** *mac-address* | **method** *method* | **registrations** | **session-id** *session-id* | **statistics** | **switch** *switch-number* | **details** }

## Syntax Description

| | |
|---|---|
| **database** | (Optional) Displays session data stored in the session database. This allows you to see information such as the VLAN ID, which is not cached internally. A warning message is displayed if the data stored in the session database does not match the internally cached data. |
| **method** | (Optional) Displays information about subscriber sessions using one of the following authentication methods:<br>• dot1x: IEEE 802.1X authentication method.<br>• mab: MAC authentication bypass (MAB) method.<br>• webauth: Web authentication method.<br>If you specify a method, you can also specify an interface. |
| **brief** | (Optional) Displays brief information about authentication sessions. |
| **cache** | (Optional) Displays Session Manager cache information |
| **event-logging** | (Optional) Displays event logs. |
| **fqdn** | (Optional) Displays the FQDN configurations. |
| **history** | (Optional) Displays historic information. |
| **info** | (Optional) Displays brief information about all sessions. |
| **interface** | (Optional) Displays information about subscriber sessions that match the specified client interface type. To display the valid keywords and arguments for interfaces, use the question mark (**?**) online help function. |
| **mac** | (Optional) Displays information about subscriber sessions with the specified client MAC address. |
| **session-id** | (Optional) Displays information about subscriber sessions with the specified client session identifier. |
| **registrations** | (Optional) Displays information about all registered session manager clients including the registered authentication methods. |
| **statistics** | (Optional) Displays information about authentication session statistics. |

| | |
|---|---|
| **details** | (Optional) Displays detailed information about each session instead of displaying a single-line summary. |

**Command Default**

Information about Session Aware Networking sessions is displayed.

**Command Modes**

Privileged EXEC (#)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Dublin 17.11.1 | This command was modified. The **info** keyword was introduced for this command. |

**Usage Guidelines**

If you enter the **show access-session** command without any keywords or arguments, the information is displayed for all the sessions on the switch. When you specify an identifier, information is displayed for only those sessions that match the identifier.

**Examples**

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

```
Device# show access-session
Interface               MAC Address    Method  Domain  Status Fg  Session ID
-----------------------------------------------------------------------------------------
Te1/0/23                000c.2946.8752 mab     DATA    Auth        910C140B00003E9AE7A39739
Gi3/0/6                 0015.0100.0001 dot1x   DATA    Auth        910C140B00003E9CE7A3DEC1

Session count = 2

Key to Session Events Blocked Status Flags:

  A - Applying Policy (multi-line status for details)
  D - Awaiting Deletion
  F - Final Removal in progress
  I - Awaiting IIF ID allocation
  P - Pushed Session
  R - Removing User Profile (multi-line status for details)
  U - Applying User Profile (multi-line status for details)
  X - Unknown Blocker
```

The following is sample output from the **show access-session** command with the **interface** keyword:

```
Device# show access-session interface TenGigabitEthernet1/0/23
Interface               MAC Address    Method  Domain  Status Fg  Session ID
-----------------------------------------------------------------------------------------
Te1/0/23                000c.2946.8752 mab     DATA    Auth        910C140B00003E9AE7A39739

Key to Session Events Blocked Status Flags:

  A - Applying Policy (multi-line status for details)
  D - Awaiting Deletion
  F - Final Removal in progress
  I - Awaiting IIF ID allocation
  P - Pushed Session
  R - Removing User Profile (multi-line status for details)
  U - Applying User Profile (multi-line status for details)
```

```
  X - Unknown Blocker

Runnable methods list:
  Handle  Priority  Name
      13         5  dot1xSup
       1         5  dot1x
       2        10  webauth
      14        15  mab
```

The following is sample output from the **show access-session** command with the **registrations** keyword:

```
Device# show access-session interface registrations
Clients registered with the Session Manager:

Handle           Priority         Name
---------------------------------------------------------------------
3                0                SVM
4                0                LWA_GUESTUSER_LOGOUT_CALLBACK_METHO
5                0                linksec
6                0                BM
7                0                SM Reauth PLUG-IN
8                0                Tag
9                0                EPM Plugin VLAN
10               0                EPM PLUGIN INTE
11               0                SM Accounting Feature
12               0                AAA LOCAL EAP
15               0                Device_Classifier
16               0                eEdge IAL SM
14               15               mab
13               5                dot1xSup
2                10               webauth
1                5                dot1x
```

The following is sample output from the **show access-session** command with the **mac** keyword:

```
Device# show access-session mac address details
Interface:  TenGigabitEthernet1/0/23
               IIF-ID:  0x1D61C9FE
          MAC Address:  000c.2946.8752
         IPv6 Address:  Unknown
         IPv4 Address:  192.0.2.1
            User-Name:  00-0C-29-46-87-52
          Device-type:  VMWare-Device
          Device-name:  VMWARE, INC.
               Status:  Authorized
               Domain:  DATA
       Oper host mode:  multi-auth
     Oper control dir:  both
      Session timeout:  600s (server), Remaining: 538s
       Timeout action:  Reauthenticate
    Common Session ID:  910C140B00003E98E787C749
      Acct Session ID:  Unknown
               Handle:  0x9e000ec3
       Current Policy:  MAB


Server Policies:
      Session-Timeout: 600 sec
     URL Redirect ACL: web_acl
         URL Redirect:
https://11.19.0.199:8443/portal/gateway?sessionId=910C140B00003E98E787C749&portal=d06bc251-f644-4fc3-b09f-dae9bd8a86d5&action=cwa&token=00227d56310f2862c2ac6eb79d5246db
```

```
Method status list:
       Method           State
          mab           Authc Success
```

The following is sample output from the **show access-session** command with the **info** keyword:

**Note** This **show access-session info** command is applicable to Identity Based Networking Services 2.0:

```
Device# show access-session interface info
Interface    MAC Address    M:D:S    VLAN     IPv4        Policy               User-Role
-----------------------------------------------------------------------------------------
Te1/0/23     000c.2946.8752 Mab:D:AZ  UA      192.0.2.1   MAB                  UA
Gi3/0/6      0015.0100.0001 D1x:D:AZ  UA      192.0.2.2   Dot1x                ABCDEFGH..

Session count = 2

Key to session Method Domain Status:

  M - Method :
  D1x - 802.1x, Mab - Mab, Web - WebAuth, N/A - Not Applicable
  D - Domain:
  D - Data, V - Voice, U - Unknown
  S - Status:
  AZ - Authorized, UZ - UnAuthorized
  UA - Un-Available
```

The table below describes the significant fields shown in the displays.

***Table 10: show access-session Field Descriptions***

| Field | Description |
|---|---|
| Interface | The interface on which the client is connected. |
| MAC Address | The MAC address of the client. |
| Method | The AAA authentication method. |
| Domain | The name of the domain, either DATA or VOICE. |
| Status | The status of the authentication session. |
| M:D:S | The consolidated column for Method, Domain, and Status. |

| | |
|---|---|
| Fg | These status flags indicate that events are temporarily blocked from being processed in a session, usually because an asynchronous action is in progress. A transient block, from less than a second to a few seconds maximum, is to be expected; a session that remains blocked for more than a few seconds indicates an issue.<br>All flags are mutually exclusive except P, which can be displayed with any other flag.<br>Key to Session Events Blocked Status Flags:<br>• A - Applying Policy (multiline status for details); A policy action (event) is being carried out and involves asynchronous processing that is in progress. Use the **details** keyword to see the name of the event being processed.<br>• D - Awaiting Deletion—Session deletion has begun. One or more asynchronous actions are currently in progress (either retrieving accounting data from the platform or deleting the IF ID).<br>• F - Final Removal in progress: The D stage is over, but the session has not been deleted yet.<br>• I - Awaiting IIF ID allocation: The IIF ID is a system-wide identifier for a session or any other object the platform must know about. The platform must have the IIF ID before proceeding.<br>• P - Pushed Session: Indicates that the session was authenticated earlier and pushed from the wireless controller module (WCM). The session manager only tracks the session; it does not perform authentication. This is for wireless sessions only. It is a permanent flag, and can be displayed with other flags.<br>• R - Removing User Profile (multiline status for details): User profile is being removed asynchronously by the enforcement policy module (EPM).<br>• U - Applying User Profile (multiline status for details): User profile is being applied asynchronously by the EPM.<br>• X - Unknown Blocker: Event is blocked for an unknown reason. |
| IPv4 | The IPv4 address of the client. |
| VLAN | The VLAN ID applied through ISE or service template. |
| Policy | The name of the policy map configured. |
| User-Role | The role of the client. |
| Handle | The context handle of clients registered with the authorization manager. |

## Related Commands

| Command | Description |
|---|---|
| **show access-session interface** *interface-name* **details** | Displays all the details of the clients of the given interface. |
| **show access-session registrations** | Displays the components registered with session manager. |

# show authentication brief

To display brief information about authentication sessions for a given interface, use the **show authentication brief** command in either user EXEC or privileged EXEC mode.

```
show authentication brief[switch{switch-number|active|standby}{R0}]
```

## Syntax Description

| | |
|---|---|
| *switch-number* | Valid values for the *switch-number* variable are from 1 to 9. |
| **R0** | Displays information about the Route Processor (RP) slot 0. |
| **active** | Specifies the active instance. |
| **standby** | Specifies the standby instance. |

## Command Modes

Privileged EXEC (#)

User EXEC (>)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

The following is a sample output from the **show authentication brief** command:

```
Device# show authentication brief

Interface  MAC Address     AuthC           AuthZ                    Fg  Uptime
-------------------------------------------------------------------------------
Gi2/0/14   0002.0002.0001  m:NA d:OK      AZ: SA-                   X     281s
Gi2/0/14   0002.0002.0002  m:NA d:OK      AZ: SA-                   X     280s
Gi2/0/14   0002.0002.0003  m:NA d:OK      AZ: SA-                   X     279s
Gi2/0/14   0002.0002.0004  m:NA d:OK      AZ: SA-                   X     278s
Gi2/0/14   0002.0002.0005  m:NA d:OK      AZ: SA-                   X     278s
Gi2/0/14   0002.0002.0006  m:NA d:OK      AZ: SA-                   X     277s
Gi2/0/14   0002.0002.0007  m:NA d:OK      AZ: SA-                   X     276s
Gi2/0/14   0002.0002.0008  m:NA d:OK      AZ: SA-                   X     276s
Gi2/0/14   0002.0002.0009  m:NA d:OK      AZ: SA-                   X     275s
Gi2/0/14   0002.0002.000a  m:NA d:OK      AZ: SA-                   X     275s
Gi2/0/14   0002.0002.000b  m:NA d:OK      AZ: SA-                   X     274s
Gi2/0/14   0002.0002.000c  m:NA d:OK      AZ: SA-                   X     274s
Gi2/0/14   0002.0002.000d  m:NA d:OK      AZ: SA-                   X     273s
Gi2/0/14   0002.0002.000e  m:NA d:OK      AZ: SA-                   X     273s
Gi2/0/14   0002.0002.000f  m:NA d:OK      AZ: SA-                   X     272s
Gi2/0/14   0002.0002.0010  m:NA d:OK      AZ: SA-                   X     272s
Gi2/0/14   0002.0002.0011  m:NA d:OK      AZ: SA-                   X     271s
Gi2/0/14   0002.0002.0012  m:NA d:OK      AZ: SA-                   X     271s
Gi2/0/14   0002.0002.0013  m:NA d:OK      AZ: SA-                   X     270s
Gi2/0/14   0002.0002.0014  m:NA d:OK      AZ: SA-                   X     270s
Gi2/0/14   0002.0002.0015  m:NA d:OK      AZ: SA-                   X     269s
```

The following is a sample output from the **show authentication brief** command for active instances:

```
Device# show authentication brief switch active R0

Interface  MAC Address     AuthC           AuthZ                    Fg  Uptime
-------------------------------------------------------------------------------
Gi2/0/14   0002.0002.0001  m:NA d:OK      AZ: SA-                   X       1s
Gi2/0/14   0002.0002.0002  m:NA d:OK      AZ: SA-                   X       0s
Gi2/0/14   0002.0002.0003  m:NA d:OK      AZ: SA-                   X     299s
Gi2/0/14   0002.0002.0004  m:NA d:OK      AZ: SA-                   X     298s
Gi2/0/14   0002.0002.0005  m:NA d:OK      AZ: SA-                   X     298s
Gi2/0/14   0002.0002.0006  m:NA d:OK      AZ: SA-                   X     297s
Gi2/0/14   0002.0002.0007  m:NA d:OK      AZ: SA-                   X     296s
Gi2/0/14   0002.0002.0008  m:NA d:OK      AZ: SA-                   X     296s
Gi2/0/14   0002.0002.0009  m:NA d:OK      AZ: SA-                   X     295s
Gi2/0/14   0002.0002.000a  m:NA d:OK      AZ: SA-                   X     295s
Gi2/0/14   0002.0002.000b  m:NA d:OK      AZ: SA-                   X     294s
Gi2/0/14   0002.0002.000c  m:NA d:OK      AZ: SA-                   X     294s
Gi2/0/14   0002.0002.000d  m:NA d:OK      AZ: SA-                   X     293s
Gi2/0/14   0002.0002.000e  m:NA d:OK      AZ: SA-                   X     293s
Gi2/0/14   0002.0002.000f  m:NA d:OK      AZ: SA-                   X     292s
Gi2/0/14   0002.0002.0010  m:NA d:OK      AZ: SA-                   X     292s
Gi2/0/14   0002.0002.0011  m:NA d:OK      AZ: SA-                   X     291s
Gi2/0/14   0002.0002.0012  m:NA d:OK      AZ: SA-                   X     291s
Gi2/0/14   0002.0002.0013  m:NA d:OK      AZ: SA-                   X     290s
Gi2/0/14   0002.0002.0014  m:NA d:OK      AZ: SA-                   X     290s
Gi2/0/14   0002.0002.0015  m:NA d:OK      AZ: SA-                   X     289s
Gi2/0/14   0002.0002.0016  m:NA d:OK      AZ: SA-                   X     289s
```

The following is a sample output from the **show authentication brief** command for standby instances:

```
Device# show authentication brief switch standby R0

No sessions currently exist
```

The table below describes the significant fields shown in the displays.

***Table 11: show authentication brief Field Descriptions***

| Field | Description |
|---|---|
| Interface | The type and number of the authentication interface. |
| MAC Address | The MAC address of the client. |
| AuthC | Indicates authentication status. |
| AuthZ | Indicates authorization status. |

| Field | Description |
|---|---|
| Fg | Flag indicates the current status. The valid values are:<br>• A—Applying policy (multi-line status for details)<br>• D—Awaiting removal<br>• F—Final removal in progress<br>• I—Awaiting IIF ID allocation<br>• P—Pushed session<br>• R—Removing user profile (multi-line status for details)<br>• U—Applying user profile (multi-line status for details)<br>• X—Unknown blocker |
| Uptime | Indicates the duration since which the session came up |

# show authentication history

To display the authenticated sessions alive on a device, use the **show authentication history** command in user EXEC or privileged EXEC mode.

**show authentication history** [**min-uptime** *seconds*]

## Syntax Description

| | |
|---|---|
| **min-uptime** *seconds* | (Optional) Displays sessions within the minimum uptime. The range is from 1 through 4294967295 seconds. |

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **show authentication history** command to display the authenticated sessions alive on the device.

The following is sample output from the **show authentication history** command:

```
Device# show authentication history

Interface  MAC Address     Method  Domain  Status  Uptime
Gi3/0/2    0021.d864.07c0  dot1x   DATA    Auth    38s

Session count = 1
```

# show authentication sessions

To display information about current Auth Manager sessions, use the **show authentication sessions** command.

**show authentication sessions** [**database**] [**handle** *handle-id* [**details**]] [**interface** *type number* [**details**] [**mac** *mac-address* [**interface** *type number*] [**method** *method-name* [**interface** *type number* [**details**] [**session-id** *session-id* [**details**]]

## Syntax Description

| | |
|---|---|
| **database** | (Optional) Shows only data stored in session database. |
| **handle** *handle-id* | (Optional) Specifies the particular handle for which Auth Manager information is to be displayed. |
| **details** | (Optional) Shows detailed information. |
| **interface** *type number* | (Optional) Specifies a particular interface type and number for which Auth Manager information is to be displayed. |
| **mac** *mac-address* | (Optional) Specifies the particular MAC address for which you want to display information. |
| **method** *method-name* | (Optional) Specifies the particular authentication method for which Auth Manager information is to be displayed. If you specify a method (**dot1x**, **mab**, or **webauth**), you may also specify an interface. |
| **session-id** *session-id* | (Optional) Specifies the particular session for which Auth Manager information is to be displayed. |

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **show authentication sessions** command to display information about all current Auth Manager sessions. To display information about specific Auth Manager sessions, use one or more of the keywords.

This table shows the possible operating states for the reported authentication sessions.

***Table 12: Authentication Method States***

| State | Description |
|---|---|
| Not run | The method has not run for this session. |
| Running | The method is running for this session. |
| Failed over | The method has failed and the next method is expected to provide a result. |

| State | Description |
|---|---|
| Success | The method has provided a successful authentication result for the session. |
| Authc Failed | The method has provided a failed authentication result for the session. |

This table shows the possible authentication methods.

***Table 13: Authentication Method States***

| State | Description |
|---|---|
| dot1x | 802.1X |
| mab | MAC authentication bypass |
| webauth | web authentication |

The following example shows how to display all authentication sessions on the device:

```
Device# show authentication sessions

Interface    MAC Address     Method   Domain   Status         Session ID
Gi1/0/48     0015.63b0.f676  dot1x    DATA     Authz Success  0A3462B1000000102983C05C
Gi1/0/5      000f.23c4.a401  mab      DATA     Authz Success  0A3462B10000000D24F80B58
Gi1/0/5      0014.bf5d.d26d  dot1x    DATA     Authz Success  0A3462B10000000E29811B94
```

The following example shows how to display all authentication sessions on an interface:

```
Device# show authentication sessions interface gigabitethernet2/0/47

            Interface:  GigabitEthernet2/0/47
          MAC Address:  Unknown
           IP Address:  Unknown
               Status:  Authz Success
               Domain:  DATA
      Oper host mode:  multi-host
    Oper control dir:  both
       Authorized By:  Guest Vlan
         Vlan Policy:  20
      Session timeout:  N/A
         Idle timeout:  N/A
    Common Session ID:  0A3462C8000000000002763C
      Acct Session ID:  0x00000002
               Handle:  0x25000000
Runnable methods list:
       Method   State
       mab      Failed over
       dot1x    Failed over
----------------------------------------
            Interface:  GigabitEthernet2/0/47
          MAC Address:  0005.5e7c.da05
           IP Address:  Unknown
            User-Name:  00055e7cda05
               Status:  Authz Success
```

```
               Domain:  VOICE
      Oper host mode:  multi-domain
    Oper control dir:  both
       Authorized By:  Authentication Server
     Session timeout:  N/A
        Idle timeout:  N/A
  Common Session ID:  0A3462C8000000010002A238
    Acct Session ID:  0x00000003
               Handle:  0x91000001
Runnable methods list:
      Method   State
      mab      Authc Success
      dot1x    Not run
```

# show cisp

To display Client Information Signaling Protocol (CISP) information for a specified interface, use the **show cisp** command in privileged EXEC mode.

**show cisp** { [**clients** | **interface** *interface-id*] | **registrations** | **summary**}

## Syntax Description

| | |
|---|---|
| **clients** | (Optional) Display CISP client details. |
| **interface** *interface-id* | (Optional) Display CISP information about the specified interfac channels. |
| **registrations** | Displays CISP registrations. |
| **summary** | (Optional) Displays CISP summary. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

The following is sample output from the **show cisp interface** command:

```
Device# show cisp interface fastethernet 0/1/1

CISP not enabled on specified interface
```

The following is sample output from the **show cisp registration** command:

```
Device# show cisp registrations

Interface(s) with CISP registered user(s):
-----------------------------------------
Fa1/0/13
Auth Mgr (Authenticator)
Gi2/0/1
Auth Mgr (Authenticator)
Gi2/0/2
Auth Mgr (Authenticator)
Gi2/0/3
Auth Mgr (Authenticator)
Gi2/0/5
Auth Mgr (Authenticator)
Gi2/0/9
Auth Mgr (Authenticator)
Gi2/0/11
Auth Mgr (Authenticator)
Gi2/0/13
Auth Mgr (Authenticator)
Gi3/0/3
Gi3/0/5
```

```
Gi3/0/23
```

**Related Commands**

| Command | Description |
|---|---|
| **cisp enable** | Enables CISP. |
| **dot1x credentials** *profile* | Configures a profile on a supplicant device. |

# show device-tracking capture-policy

To display the rules that the system pushes to the hardware (forwarding layer), enter the **show device-tracking capture-policy** command in privileged EXEC mode. These rules determine which packets are punted to SISF for further action. These rules are a translation of the policy that is applied to the interface or VLAN.

**show device-tracking capture-policy** [ **interface** *inteface_type_no* | **vlan** *vlan_id* ]

## Syntax Description

| | |
|---|---|
| **interface** *inteface_type_no* | Displays message capture policy information for the interface you specify. Enter an interface type and number.<br>Use the question mark (?) online help function to display the types of interfaces on the device. |
| **vlan** *vlan_id* | Displays message capture policy information for the VLAN ID you specify. The valid value range is from 1 to 4095. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The output of this command is used by the technical support team, for troubleshooting.

### Examples

The following is sample output from the **show device-tracking capture-policy** command:

```
Device# show device-tracking capture-policy interface tengigabitethernet1/0/1

    HW Target Te1/0/1 HW policy signature 0001DF9F policies#:1 rules 14 sig 0001DF9F
        SW policy sisf-01 feature Device-tracking - Active

        Rule DHCP4 CLIENT Protocol UDP mask 00000400 action PUNT match1 0 match2 67#feat:1
            feature Device-tracking
        Rule DHCP4 SERVER SOURCE Protocol UDP mask 00001000 action PUNT match1 0 match2
68#feat:1
            feature Device-tracking
        Rule DHCP4 SERVER Protocol UDP mask 00000800 action PUNT match1 67 match2 0#feat:1
            feature Device-tracking
        Rule ARP Protocol IPV4 mask 00004000 action PUNT match1 0 match2 0#feat:1
            feature Device-tracking
        Rule DHCP SERVER SOURCE Protocol UDP mask 00000200 action PUNT match1 0 match2
546#feat:1
            feature Device-tracking
        Rule DHCP CLIENT Protocol UDP mask 00000080 action PUNT match1 0 match2 547#feat:1
            feature Device-tracking
        Rule DHCP SERVER Protocol UDP mask 00000100 action PUNT match1 547 match2 0#feat:1
            feature Device-tracking
        Rule RS Protocol ICMPV6 mask 00000004 action PUNT match1 133 match2 0#feat:1
            feature Device-tracking
        Rule RA Protocol ICMPV6 mask 00000008 action PUNT match1 134 match2 0#feat:1
```

```
    feature Device-tracking
Rule NS Protocol ICMPV6 mask 00000001 action PUNT match1 135 match2 0#feat:1
    feature Device-tracking
Rule NA Protocol ICMPV6 mask 00000002 action PUNT match1 136 match2 0#feat:1
    feature Device-tracking
Rule REDIR Protocol ICMPV6 mask 00000010 action PUNT match1 137 match2 0#feat:1
    feature Device-tracking
Rule DAR Protocol ICMPV6 mask 00008000 action PUNT match1 157 match2 0#feat:1
    feature Device-tracking
Rule DAC Protocol ICMPV6 mask 00010000 action PUNT match1 158 match2 0#feat:1
    feature Device-tracking
```

# show device-tracking counters

To display information about the number of broadcast, multicast, bridged, unicast, probe, dropped device-tracking messages and faults received on an interface or VLAN or both, enter the **show device-tracking counters** command in privileged EXEC mode. Where applicable, the messages are categorized by protocol. The list of protocols include Address Resolution Protocol (ARP), Neighbor Discovery Protocol (NDP), DHCPv6, DHCPv4, Address Collision Detection (ACD), and Duplicate Address Detection (DAD).

**show device-tracking counters** [ **all** | **interface** *inteface_type_no* | **vlan** *vlan_id* ]

## Syntax Description

| | |
|---|---|
| **all** | Displays information for all interfaces and VLANs on the device where a policy is attached. |
| **interface** *inteface_type_no* | Displays information for the specified interface. Enter an interface type and number.<br>Use the question mark (?) online help function to display the types of interfaces on the device. |
| **vlan** *vlan_id* | Displays information for the VLAN ID you specify. The range is from 1 to 4095. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

When you enter the **show device-tracking counters**command, you must enter one of the keywords that follow, that is, **all**, or **interface** *inteface_type_no* , or **vlan** *vlan_id* .

If you specify an interface or VLAN where a policy is not attached, the following message is displayed: `% no ipv6 snooping policy attached on <interface number or VLAN ID>`

### Examples

The following is sample output from the**show device-tracking counters** command. Information relating to a particular VLAN (VLAN 10) is displayed here:

```
Device# show device-tracking counters vlan 10
Received messages on vlan 10   :
Protocol        Protocol message
NDP             RA[2479] NS[1757] NA[2794]
DHCPv6
ARP             REP[878]
DHCPv4
ACD&DAD         --[3]

Received Broadcast/Multicast messages on vlan 10   :
Protocol        Protocol message
NDP             RA[2479] NS[3] NA[5]
DHCPv6
```

```
ARP             REP[1]
DHCPv4

Bridged messages from vlan 10   :
Protocol        Protocol message
NDP             RA[1238] NS[1915] NA[878]
DHCPv6
ARP             REQ[877]
DHCPv4
ACD&DAD         --[1]

Broadcast/Multicast converted to unicast messages from vlan 10   :
Protocol        Protocol message
NDP
DHCPv6
ARP
DHCPv4
ACD&DAD

Probe message on vlan 10   :
Type            Protocol message
PROBE_SEND      NS[1037] REQ[877]
PROBE_REPLY     NA[1037] REP[877]

Limited Broadcast to Local message on vlan 10   :
Type            Protocol message
NDP
DHCPv6
ARP
DHCPv4

Dropped messages on vlan 10   :
Feature             Protocol Msg [Total dropped]
Device-tracking:    NDP      RA  [1241]
                    reason:  Packet not authorized on port [1241]

                             NS  [2]
                    reason:  Silent drop [2]

                             NA  [1039]
                    reason:  Silent drop [1037]
                    reason:  Packet accepted but not forwarded [2]

                    ARP      REP [878]
                    reason:  Silent drop [877]
                    reason:  Packet accepted but not forwarded [1]

ACD&DAD:            --       --  [2]


Faults on vlan 10   :
```

# show device-tracking database

To display details of the binding table database, enter the **show device-tracking database** command in privileged EXEC mode.

**show device-tracking database** [ **address** { *hostname_address* | **all** } [ **interface** *inteface_type_no* ] [ **vlanid** *vlan* ] [ **details** ] | **details** | **interface** *inteface_type_no* [ **details** ] [ **vlanid** *vlan* ] | **mac** [ *48_bit_hw_add* ] [ **details** ] [ **interface** *inteface_type_no* ] [ **vlanid** *vlan* ] | **prefix** [ *prefix_address* | **all** ] [ **details** ] [ **interface** *inteface_type_no* ] | **vlanid** *vlanid* [ **details** ] ]

## Syntax Description

| | |
|---|---|
| **address** {*hostname_address* \| **all**} | Displays binding table information for a particular IP address or for all addresses |
| **interface** *inteface_type_no* | Displays binding table information for the specified interface. Enter an interface type and number.<br>Use the question mark (?) online help function to display the types of interfaces on the device. |
| **vlanid** *vlan* | Displays binding table information for the VLAN ID you specify. The valid value range is from 1 to 4095. |
| **details** | Displays detailed information. |
| **mac** | Displays binding table information for the MAC address you specify. |
| *48_bit_hw_add* | Enter a 48-bit hardware address. |
| **prefix** | Displays binding table information for the IPv6 prefix you specify. |
| *prefix_address* | Enter an IPv6 prefix. |
| **all** | Displays binding table information for all the available IPv6 prefixes. |

## Command Modes

Privileged EXEC (# )

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

### Examples

The following is sample output for the **show device-tracking database details**command. The accompanying table describes the significant fields shown in the display.

```
Device# show device-tracking database details

 Binding table configuration:
 ---------------------------
 max/box  : no limit
 max/vlan : no limit
```

```
 max/port : no limit
 max/mac  : no limit

 Binding table current counters:
 -------------------------------
 dynamic  : 5
 local    : 1
 total    : 5

 Binding table counters by state:
 ---------------------------------
 REACHABLE  : 5
      DOWN  : 1
     total  : 6

Codes: L - Local, S - Static, ND - Neighbor Discovery, ARP - Address Resolution Protocol,
DH4 - IPv4 DHCP, DH6 - IPv6 DHCP, PKT - Other Packet, API - API created
Preflevel flags (prlvl):
0001:MAC and LLA match     0002:Orig trunk            0004:Orig access
0008:Orig trusted trunk    0010:Orig trusted access   0020:DHCP assigned
0040:Cga authenticated     0080:Cert authenticated    0100:Statically assigned


    Network Layer Address   Link Layer Address   Interface  mode     vlan(prim)   prlvl
age    state      Time left   Filter   In Crimson   Client ID          Session ID
Policy (feature)
ARP 192.0.9.29              001b.4411.3ab7(S)    Te1/0/4    trunk    200 ( 200)   0003
6mn    REACHABLE  331 s       no       yes          0000.0000.0000     (unspecified)
sisf-01 (Device-tracking)
ARP 192.0.9.28              001b.4411.3ab7(S)    Te1/0/4    trunk    200 ( 200)   0003
6mn    REACHABLE  313 s       no       yes          0000.0000.0000     (unspecified)
sisf-01 (Device-tracking)
ARP 192.0.9.27              001b.4411.3ab7(S)    Te1/0/4    trunk    200 ( 200)   0003
6mn    REACHABLE  323 s       no       yes          0000.0000.0000     (unspecified)
sisf-01 (Device-tracking)
ARP 192.0.9.26              001b.4411.3ab7(S)    Te1/0/4    trunk    200 ( 200)   0003
6mn    REACHABLE  311 s       no       yes          0000.0000.0000     (unspecified)
sisf-01 (Device-tracking)
ARP 192.0.9.25              001b.4411.3ab7(S)    Te1/0/4    trunk    200 ( 200)   0003
6mn    REACHABLE  313 s       no       yes          0000.0000.0000     (unspecified)
sisf-01 (Device-tracking)
L   192.168.0.1              00a5.bf9d.0462(D)      Vl200      svi    200 ( 200)   0100
6mn    DOWN                    no       yes          0000.0000.0000     (unspecified)
sisf-01 (sisf_local)
```

*Table 14: show device-tracking database details Field Descriptions*

| Field | Description |
|---|---|
| Binding table configuration:<br>• max/box<br>• max/vlan<br>• max/port<br>• max/mac | Displays binding table settings. The values correspond with what is configured using the **device-tracking binding** command in global configuration mode.<br>• max/box: The value displayed here corresponds with the configured value for the **max-entries** *no_of_entries* keyword.<br>• max/vlan: The value displayed here corresponds with the configured value for the **vlan-limit** *no_of_entries* keyword.<br>• max/port: The value displayed here corresponds with the configured value for the **port-limit** *no_of_entries* keyword.<br>• max/mac: The value displayed here corresponds with the configured value for the **mac-limit** *no_of_entries* keyword. |
| Binding table current counters:<br>• dynamic<br>• local<br>• total | Displays the number of entries in the table.<br>• dynamic: Dynamic entries are created by learning events that dynamically populate the binding table.<br>• local: Local entries are automatically created when you configure an SVI on the device.<br>One of ways in which SISF uses a local entry, is in the context of polling. If polling is enabled, the SVI address is used as the source address of an ARP probe.<br>• total: The total is a sum of the dynamic, local, and static binding entries. |
| Binding table counters by state: | Displays the number of entries in each state. The state can be REACHABLE, STALE, DOWN. |
| Codes | Clarifies abbreviations that are used to signify learning events.<br>The first column of a binding entry uses an abbreviated code, which tells you about the learning event that resulted in creation of that binding entry. |

| Field | Description |
|---|---|
| Preflevel flags (prlvl) | A list of preference level number codes and clarification for what the number codes in the `prlvl` column of the binding table mean.<br>The codes signify a broad classification and multiple codes can apply to an entry. What is displayed in the `prlvl` column is a sum of these number codes and signifies a corresponding preference level.<br>For example if an ARP entry (preference code: 0001) is learned from an access interface (preference code: 0004), the value displayed in the `prlvl` column is "0005".<br>1 is the lowest preference level, and 100 is the highest.<br>A binding entry with a higher preference is given preference in case of a collision. For example, if the same entry is seen on two different interfaces, the value in the `prlvl` column, determines which entry is retained. |
| Network Layer Address | The IP address of the host from which a packet is received. |
| Link Layer Address | The MAC address of the host. |
| Mode | Displays one of the following values: "invalid", "unsupp", "access", "trunk", "vpc", "svi", "virtual", "pseudowire", "unkn", "bdi", "pseudoport". |
| vlan(prim) | The host's VLAN ID |
| prlvl | A value between 1 and 100 is displayed, with 1 having the lowest preference level, and 100 having the highest preference level.<br>See `Preflevel flags` above to know what the value displayed here means. |
| age | The total age of the entry in seconds (s) or minutes (mn) since the the last time the entry was refreshed. When it is refreshed (sign-of-life from the host), this value is reset. |
| state | The current state of an entry, which can be one of the stable or transitional states.<br>Stable state values are: REACHABLE, DOWN, and STALE,<br>Transitional states values are: VERIFY, INCOMPLETE, and TENTATIVE. |

| Field | Description |
|---|---|
| Time left | Displays the amount of time left until the next action in the current state. |
| In Crimson | A *yes* or *no* value which indicates if the entry has been added to another database. The information is then used by other applications, like Cisco DNA Center.<br>Typically, all the entries that are in a binding table are also added to this database.<br>This is used by the technical support team, for troubleshooting and to diagnose a problem. |
| Client ID | This field is applicable only to virtual machines (VMs) in Cisco Software-Defined Access (SDA) deployments.<br>It refers to the actual MAC address of a VM in a bridged networking mode, where the hosting device is a wireless client with a non-promiscuous network interface (NIC). |
| Session ID | This field is applicable only to VMs in SDA deployments.<br>It refers to an access session ID for a VM in a bridged networking mode. Each Session ID is associated with a Client ID. SISF maintains this association and transfers it along as the VM roams or moves across fabric edges in an SDA setup. |
| Policy (feature) | Displays the name of the policy applied to the interface or VLAN.<br>The "(feature)" displayed is always "Device-tracking", because only SISF-based device-tracking supports the creation of binding entries. |

# show device-tracking events

To display SISF binding table-related events, enter the **show device-tracking events** command in privileged EXEC mode. The types of events that are displayed includes the creation of binding table entries and all updates to an entry. Updates may be state changes, or, changes in the MAC, VLAN, or interface information for an entry.

**show device-tracking events**

## Syntax Description

This command has no arguments or keywords.

## Command Default

SISF binding table events are displayed.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The output of this command is used by the technical support team, for troubleshooting.

### Examples

The following is sample output for the **show device-tracking events** command. It shows you the kind of binding table events that the system logs:

```
Device# show device-tracking events
 [Wed Mar 23 19:08:33.000] SSID 0 FSM Feature Table running for event ACTIVE_REGISTER in
state CREATING
 [Wed Mar 23 19:08:33.000] SSID 0 Transition from CREATING to READY upon event ACTIVE_REGISTER

 [Wed Mar 23 19:08:33.000] SSID 1 FSM Feature Table running for event ACTIVE_REGISTER in
state CREATING
 [Wed Mar 23 19:08:33.000] SSID 1 Transition from CREATING to READY upon event ACTIVE_REGISTER

 [Wed Mar 23 19:09:25.000] SSID 0 FSM sisf_mac_fsm running for event MAC_TENTV in state
MAC-CREATING
 [Wed Mar 23 19:09:25.000] SSID 0 Transition from MAC-CREATING to MAC-TENTATIVE upon event
 MAC_TENTV
 [Wed Mar 23 19:09:25.000] SSID 1 Created Entry origin IPv4 ARP MAC 00a5.bf9c.e051 IPV4
10.0.0.1
 [Wed Mar 23 19:09:25.000] SSID 0 FSM sisf_mac_fsm running for event MAC_VERIFIED in state
 MAC-TENTATIVE
 [Wed Mar 23 19:09:25.000] SSID 0 Transition from MAC-TENTATIVE to MAC-REACHABLE upon event
 MAC_VERIFIED
 [Wed Mar 23 19:09:25.000] SSID 1 FSM Binding table running for event VALIDATE_LLA in state
 CREATING
 [Wed Mar 23 19:09:25.000] SSID 1 FSM Binding table running for event SET_TENTATIVE in state
 CREATING
 [Wed Mar 23 19:09:25.000] SSID 1 Transition from CREATING to TENTATIVE upon event
SET_TENTATIVE
 [Wed Mar 23 19:09:25.000] SSID 1 Entry State changed origin IPv4 ARP MAC 00a5.bf9c.e051
IPV4 10.0.0.1
```

```
 [Wed Mar 23 20:07:27.000] SSID 0 FSM sisf_mac_fsm running for event MAC_DELETE_NOS in state
 MAC-REACHABLE
 [Wed Mar 23 20:07:27.000] SSID 0 Transition from MAC-REACHABLE to MAC-NONE upon event
MAC_DELETE_NOS
 [Wed Mar 23 20:07:27.000] SSID 1 Transition from REACHABLE to NONE upon event DELETE
```

# show device-tracking features

To display the device-tracking features that are enabled, enter the **show device-tracking features** command in privileged EXEC mode. The "features" include SISF-based device-tracking, and security features like IPv6 RA Guard, IPv6 DHCP Guard, Layer 2 DHCP Relay, and so on, that use SISF.

**show device-tracking features**

## Syntax Description

This command has no arguments or keywords.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

### Examples

The following is sample output for the **show device-tracking features** command.

```
Device# show device-tracking features
Feature name   priority state
Device-tracking   128   READY
Source guard       32   READY
```

# show device-tracking messages

To display a list of device-tracking related activities, enter the **show device-tracking messages** command in privileged EXEC mode.

**show device-tracking messages** [ **detailed** *no_of_messages* ]

## Syntax Description

| | |
|---|---|
| **detailed** *no_of_messages* | Displays a more detailed format of the list of device-tracking messages. Enter a value between 1 and 255, to specify the number of messages that must be displayed in a detailed format. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

The following is sample output for the **show device-tracking messages** command. The summarized and detailed versions of the output are displayed:

```
Device# show device-tracking messages
 [Wed Mar 23 19:09:25.000] VLAN 1, From Te1/0/2 MAC 00a5.bf9c.e051: ARP::REP, 10.0.0.1,
 [Wed Mar 23 20:03:22.000] VLAN 1, From Te1/0/2 MAC 00a5.bf9c.e051: ARP::REP, 10.0.0.1,


Device# show device-tracking messages detailed 255
 [Wed Mar 23 19:09:25.000] VLAN 1, From Te1/0/2 seclvl [guard], MAC 00a5.bf9c.e051: ARP::REP,

 1 addresses advertised:
    IPv6 addr: 10.0.0.1,

 [Wed Mar 23 20:03:22.000] VLAN 1, From Te1/0/2 seclvl [guard], MAC 00a5.bf9c.e051: ARP::REP,

 1 addresses advertised:
    IPv6 addr: 10.0.0.1,
```

# show device-tracking policies

To display *all* the device-tracking policies on the device, enter the **show device-tracking policies** command in privileged EXEC mode.

**show device-tracking policies** [ **details** | **interface** *interface_type_no* [ **details** ] | **vlan** *vlanid* ]

## Syntax Description

| | |
|---|---|
| **details** | Displays information about the policy targets and policy parameters of all device-tracking policies on the device |
| **interface** *interface_type_no* | Displays all policies applied to the the specified interface. Enter an interface type and number.<br>Use the question mark (?) online help function to display the types of interfaces on the device. |
| **vlan** *vlanid* | Displays all policies applied to the the specified VLAN. The valid value range is from 1 to 4095. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

### Examples

The following is sample output for the **show device-tracking policies** command with the **details** keyword. It shows that there is only one policy on the device. It shows the target to which the policy is applied and the policy parameters.

```
Device# show device-tracking policies details

Target               Type  Policy               Feature        Target range
Te1/0/1              PORT  sisf-01              Device-tracking vlan all

Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  tracking enable
Policy sisf-01 is applied on the following targets:
Target               Type  Policy               Feature        Target range
Te1/0/1              PORT  sisf-01              Device-tracking vlan all
```

# show device-tracking policy

To display information about a particular policy, enter the **show device-tracking policy** command in privileged EXEC mode. Displayed information includes the list of targets to which the policy is applied, and policy parameters.

**show device-tracking policy** *policy_name*

## Syntax Description

| | |
|---|---|
| *policy_name* | Enter the name of the policy. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

### Examples

The following is sample output for the **show device-tracking policy** command. Details of policy `sisf-01` are displayed.

```
Device# show device-tracking policy sisf-01
Device-tracking policy sisf-01 configuration:
  security-level guard
  device-role node
  gleaning from Neighbor Discovery
  gleaning from DHCP6
  gleaning from ARP
  gleaning from DHCP4
  NOT gleaning from protocol unkn
  tracking enable
Policy sisf-01 is applied on the following targets:
Target               Type  Policy               Feature        Target range
Te1/0/1              PORT  sisf-01              Device-tracking vlan all
```

# show dot1x

To display IEEE 802.1x statistics, administrative status, and operational status for a device or for the specified port, use the **show dot1x** command in user EXEC or privileged EXEC mode.

**show dot1x** [**all** [**count** | **details** | **statistics** | **summary**]] [**interface** *type number* [**details** | **statistics**]] [**statistics**]

## Syntax Description

| | |
|---|---|
| **all** | (Optional) Displays the IEEE 802.1x information for all interfaces. |
| **count** | (Optional) Displays total number of authorized and unauthorized clients. |
| **details** | (Optional) Displays the IEEE 802.1x interface details. |
| **statistics** | (Optional) Displays the IEEE 802.1x statistics for all interfaces. |
| **summary** | (Optional) Displays the IEEE 802.1x summary for all interfaces. |
| **interface** *type number* | (Optional) Displays the IEEE 802.1x status for the specified port. |

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

The following is sample output from the **show dot1x all** command:

```
Device# show dot1x all

Sysauthcontrol              Enabled
Dot1x Protocol Version            3
```

The following is sample output from the **show dot1x all count** command:

```
Device# show dot1x all count

Number of Dot1x sessions
------------------------------
Authorized Clients        = 0
UnAuthorized Clients      = 0
Total No of Client        = 0
```

The following is sample output from the **show dot1x all statistics** command:

```
Device# show dot1x statistics
```

```
Dot1x Global Statistics for
-------------------------------------------
RxStart = 0     RxLogoff = 0    RxResp = 0      RxRespID = 0
RxReq = 0       RxInvalid = 0   RxLenErr = 0
RxTotal = 0

TxStart = 0     TxLogoff = 0    TxResp = 0
TxReq = 0       ReTxReq = 0     ReTxReqFail = 0
TxReqID = 0     ReTxReqID = 0   ReTxReqIDFail = 0
TxTotal = 0
```

# show eap pac peer

To display stored Protected Access Credentials (PAC) for Extensible Authentication Protocol (EAP) Flexible Authentication via Secure Tunneling (FAST) peers, use the **show eap pac peer** command in privileged EXEC mode.

**show eap pac peer**

## Syntax Description

This command has no arguments or keywords.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

The following is sample output from the **show eap pac peers** command:

```
Device# show eap pac peers

No PACs stored
```

## Related Commands

| Command | Description |
|---|---|
| **clear eap sessions** | Clears EAP session information for the device or for the |

# show fqdn

To display the resolved IP addresses for the fully qualified domain name (FQDN) entries in the local cache database, use the **show fqdn** command in either user EXEC mode or privileged EXEC mode.

**show fqdn** { **database** [ **dump** | **fqdn** *name* [ **ipv4** *address* | **ipv6** *address* ] | **statistics** ] | **packet statistics** | **summary** }

## Syntax Description

| | |
|---|---|
| **database** | Displays the FQDN-to-IP address bindings, along with the time-to-live (TTL) for those bindings, and the status of the bindings. |
| **dump** | (Optional) Displays all the cached and transient FQDN entries from the database. |
| **fqdn** *name* | (Optional) Displays cached entries from the database for the specified FQDN. |
| **ipv4** *address* | (Optional) Displays cached entries from the database for the specified IP address. |
| **ipv6** *address* | (Optional) Displays cached entries from the database for the specified IPv6 address. |
| **statistics** | (Optional) Displays FQDN database statistics. |
| **packet statistics** | Displays FQDN packet statistics. |
| **summary** | Displays FQDN TTL timeout factor and a summary of the registered FQDNs. |

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Bengaluru 17.5.1 | This command was introduced. |
| Cisco IOS XE Bengaluru 17.6.1 | This command was modified. The **ipv6** keyword was introduced. |

## Examples

The following is a sample output from the **show fqdn summary** command:

```
Device# show fqdn summary

Registered FQDNs:
 FQDN: host1.9k.com
 Registered Module IDs: FQDN ACL Mgr,
 FQDN: host2.9k.com
 Registered Module IDs: FQDN ACL Mgr,
 FQDN: host3.9k.com
 Registered Module IDs: FQDN ACL Mgr,
 FQDN: host4.9k.com
 Registered Module IDs: FQDN ACL Mgr,
```

The following is a sample output from the **show fqdn database** command:

```
Device# show fqdn database
```

```
====================== MAIN DB ======================
FQDN Name: example.cisco.com
IP Address Type TTL
10.11.224.132 IPv4 313/370
10.11.112.150 IPv4 313/370
```

The following is a sample output from the **show fqdn packet statistics** command:

```
Device# show fqdn packet statistics

Total DNS Pkts Rcvd : 676
IPv4 DNS Pkts Rcvd : 676
IPv6 DNS Pkts Rcvd : 0
Total DNS Pkts for registered FQDNs : 2
Total DNS Pkts with induced latency : 2
DNS Packet Latency:
Min : 34 ms
Max : 3002 ms
Avg : 1518 ms
Total DNS Pkts Injected : 676
DNS Pkts Injected by Ack : 2
DNS Pkts with parse error : 0
DNS Pkts with no answer : 674
DNS Pkts with no A/AAAA Record : 0
DNS Pkts with FQDN not registered : 0
DNS Pkts with FQDN already cached : 0
Total DNS Pkts Dropped by Nack : 0
Average Input rate - 1 min : 11 pps
Average Input rate - 5 min : 11 pps
Average Input rate - 1 hr : 11 pps
```

## Related Commands

| Command | Description |
|---|---|
| **show ip access-lists** | Displays the contents of all the current IP access lists. |
| **show ipv6 access-lists** | Displays the contents of all the current IPv6 access lists. |

# show ip access-lists

To display the contents of all current IP access lists, use the **show ip access-lists** command in user EXEC or privileged EXEC modes.

**show ip access-lists** [ *access-list-number access-list-number-expanded-range access-list-name* | **dynamic** [*dynamic-access-list-name*] | **interface** *name number* [ **in** | **out** ] ]

## Syntax Description

| | |
|---|---|
| *access-list-number* | (Optional) Number of the IP access list to display. |
| *access-list-number-expanded-range* | (Optional) Expanded range of the IP access list to display. |
| *access-list-name* | (Optional) Name of the IP access list to display. |
| **dynamic** *dynamic-access-list-name* | (Optional) Displays the specified dynamic IP access lists. |
| **interface** *name number* | (Optional) Displays the access list for the specified interface. |
| **in** | (Optional) Displays input interface statistics. |
| **out** | (Optional) Displays output interface statistics. |

**Note** Statistics for OGACL is not supported

## Command Default

All standard and expanded IP access lists are displayed.

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **show ip access-lists** command provides output identical to the **show access-lists** command, except that it is IP-specific and allows you to specify a particular access list.

The output of the **show ip access-lists interface** command does not display dACL or ACL filter IDs. This is because the ACLs are attached to the virtual ports created by multidomain authentication for each authentication session; instead of the physical interface. To display dACL or ACL filter IDs, use the **show ip access-lists** *access-list-name* command. The *access-list-name* should be taken from the **show access-session interface** *interface-name* **detail** command output. The *access-list-name* is case sensitive.

## Examples

The following is a sample output from the **show ip access-lists** command when all access lists are requested:

```
Device# show ip access-lists

Extended IP access list 101
   deny udp any any eq nntp
   permit tcp any any
   permit udp any any eq tftp
   permit icmp any any
   permit udp any any eq domain
Role-based IP access list r1
    10 permit tcp dst eq telnet
    20 permit udp
FQDN IP access list facl
    10 permit ip host 10.1.1.1 host dynamic www.google.com
    20 permit tcp 10.10.0.0 0.255.255.255 eq ftp host dynamic www.cisco.com log
    30 permit udp host dynamic www.youtube.com any
    40 permit ip 10.3.4.0 0.0.0.255 any
Extended Resolved IP access list facl
    200000 permit tcp 10.0.0.0 0.255.255.255 eq ftp host 10.10.10.1 log
    200001 permit tcp 10.0.0.0 0.255.255.255 eq ftp host 10.10.10.2 log
    300000 permit udp host dynamic 10.11.11.11 any
    300001 permit udp host dynamic 10.11.11.12 any
    400000 permit ip 10.3.4.0 0.0.0.255 any
```

The table below describes the significant fields shown in the display.

***Table 15: show ip access-lists Field Descriptions***

| Field | Description |
|---|---|
| Extended IP access list | Extended IP access-list name/number. |
| Role-based IP access list | Role-based IP access-list name. |
| FQDN IP access list | FQDN IP access-list name. |
| Extended Resolved IP access list | Extended resolved IP access-list name. |
| deny | Packets to reject. |
| udp | User Datagram Protocol. |
| any | Source host or destination host. |
| eq | Packets on a given port number. |
| nntp | Network News Transport Protocol. |
| permit | Packets to forward. |
| dynamic | Dynamically resolves domain name. |
| tcp | Transmission Control Protocol. |
| tftp | Trivial File Transfer Protocol. |
| icmp | Internet Control Message Protocol. |
| domain | Domain name service. |

The following is a sample output from the **show ip access-lists** command when the name of a specific access list is requested:

```
Device# show ip access-lists Internetfilter

Extended IP access list Internetfilter
   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
```

The following is a sample output from the **show ip access-lists** command using the **dynamic** keyword:

```
Device# show ip access-lists dynamic CM_SF#1

Extended IP access list CM_SF#1
    10 permit udp any any eq 5060 (650 matches)
    20 permit tcp any any eq 5060
    30 permit udp any any dscp ef (806184 matches)
```

## Related Commands

| Command | Description |
|---|---|
| **deny** | Sets conditions in a named IP access list or OGACL that will deny packets. |
| **ip access-group** | Applies an ACL or OGACL to an interface or a service policy map. |
| **ip access-list** | Defines an IP access list or OGACL by name or number. |
| **object-group network** | Defines network object groups for use in OGACLs. |
| **object-group service** | Defines service object groups for use in OGACLs. |
| **permit** | Sets conditions in a named IP access list or OGACL that will permit packets. |
| **show object-group** | Displays information about object groups that are configured. |
| **show run interfaces cable** | Displays statistics on the cable modem. |

# show ip dhcp snooping statistics

To display DHCP snooping statistics in summary or detail form, use the **show ip dhcp snooping statistics** command in user EXEC or privileged EXEC mode.

**show ip dhcp snooping statistics** [**detail** ]

## Syntax Description

| | |
|---|---|
| **detail** | (Optional) Displays detailed statistics information. |

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

In a device stack, all statistics are generated on the stack's active switch. If a new active device is elected, the statistics counters reset.

The following is sample output from the **show ip dhcp snooping statistics** command:

```
Device> show ip dhcp snooping statistics

 Packets Forwarded                                = 0
 Packets Dropped                                  = 0
 Packets Dropped From untrusted ports             = 0
```

The following is sample output from the **show ip dhcp snooping statistics detail** command:

```
Device> show ip dhcp snooping statistics detail

 Packets Processed by DHCP Snooping               = 0
 Packets Dropped Because
   IDB not known                                  = 0
   Queue full                                     = 0
   Interface is in errdisabled                    = 0
   Rate limit exceeded                            = 0
   Received on untrusted ports                    = 0
   Nonzero giaddr                                 = 0
   Source mac not equal to chaddr                 = 0
   Binding mismatch                               = 0
   Insertion of opt82 fail                        = 0
   Interface Down                                 = 0
   Unknown output interface                       = 0
   Reply output port equal to input port          = 0
   Packet denied by platform                      = 0
```

This table shows the DHCP snooping statistics and their descriptions:

*Table 16: DHCP Snooping Statistics*

| DHCP Snooping Statistic | Description |
|---|---|
| Packets Processed by DHCP Snooping | Total number of packets handled by DHCP snooping, including forwarded and dropped packets. |
| Packets Dropped Because IDB not known | Number of errors when the input interface of the packet cannot be determined. |
| Queue full | Number of errors when an internal queue used to process the packets is full. This might happen if DHCP packets are received at an excessively high rate and rate limiting is not enabled on the ingress ports. |
| Interface is in errdisabled | Number of times a packet was received on a port that has been marked as error disabled. This might happen if packets are in the processing queue when a port is put into the error-disabled state and those packets are subsequently processed. |
| Rate limit exceeded | Number of times the rate limit configured on the port was exceeded and the interface was put into the error-disabled state. |
| Received on untrusted ports | Number of times a DHCP server packet (OFFER, ACK, NAK, or LEASEQUERY) was received on an untrusted port and was dropped. |
| Nonzero giaddr | Number of times the relay agent address field (giaddr) in the DHCP packet received on an untrusted port was not zero, or the **no ip dhcp snooping information option allow-untrusted** global configuration command is not configured and a packet received on an untrusted port contained option-82 data. |
| Source mac not equal to chaddr | Number of times the client MAC address field of the DHCP packet (chaddr) does not match the packet source MAC address and the **ip dhcp snooping verify mac-address** global configuration command is configured. |
| Binding mismatch | Number of times a RELEASE or DECLINE packet was received on a port that is different than the port in the binding for that MAC address-VLAN pair. This indicates someone might be trying to spoof the real client, or it could mean that the client has moved to another port on the device and issued a RELEASE or DECLINE. The MAC address is taken from the chaddr field of the DHCP packet, not the source MAC address in the Ethernet header. |
| Insertion of opt82 fail | Number of times the option-82 insertion into a packet failed. The insertion might fail if the packet with the option-82 data exceeds the size of a single physical packet on the internet. |

| DHCP Snooping Statistic | Description |
|---|---|
| Interface Down | Number of times the packet is a reply to the DHCP relay agent, but the SVI interface for the relay agent is down. This is an unlikely error that occurs if the SVI goes down between sending the client request to the DHCP server and receiving the response. |
| Unknown output interface | Number of times the output interface for a DHCP reply packet cannot be determined by either option-82 data or a lookup in the MAC address table. The packet is dropped. This can happen if option 82 is not used and the client MAC address has aged out. If IPSG is enabled with the port-security option and option 82 is not enabled, the MAC address of the client is not learned, and the reply packets will be dropped. |
| Reply output port equal to input port | Number of times the output port for a DHCP reply packet is the same as the input port, causing a possible loop. Indicates a possible network misconfiguration or misuse of trust settings on ports. |
| Packet denied by platform | Number of times the packet has been denied by a platform-specific registry. |

# show platform software dns-umbrella statistics

To display the Domain Name Syatem (DNS) umbrella statistics of a device, use the **show platform software dns-umbrella statistics** command in privileged EXEC mode.

**show platform software dns-umbrella statistics**

## Syntax Description

This command has no arguments or keywords.

## Command Modes

Privileged EXEC (>)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Examples

The following is a sample output of the **show platform software dns-umbrella statistics** command:

```
Device> enable
Device# show platform software dns-umbrella statistics

=======================================
Umbrella Statistics
=======================================
Total Packets : 7848
DNSCrypt queries : 3940
DNSCrypt responses : 0
DNS queries : 0
DNS bypassed queries(Regex) : 0
DNS responses(Umbrella) : 0
DNS responses(Other) : 3906
Aged queries : 34
Dropped pkts : 0
```

# show platform software umbrella switch F0

To display umbrella configuration of Embedded Service Processor (ESP) slot 0, use the **show platform software umbrella switch** {*switch_number* | **active** | **standby**} **F0** command in privileged EXEC mode.

**show platform software umbrella switch** {*switch_number* | **active** | **standby**} **F0** {**config** | **interface-info** | **local-domain**}

## Syntax Description

| | |
|---|---|
| **switch** {*switch_number* \| **active** \| **standby**} | Specifies the switch.<br>• *switch_number*: ID of the switch. The range is from 1 to 8.<br>• **active**: Specifies the active switch.<br>• **standby**: Specifies the standby switch. |
| **config** | Displays global configurations of ESP slot 0. |
| **interface-info** | Displays interface-related configuration of ESP slot 0. |
| **local-domain** | Displays local domain-related configuration of ESP slot 0. |

## Command Modes

Privileged EXEC (>)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Examples

The following is a sample output of the **show platform software umbrella switch active F0 config** command:

```
Device> show platform software umbrella switch active F0 config

+++ Umbrella Config +++


Umbrella feature:
-----------------
Init: Enabled
Dnscrypt: disabled

Timeout:
-----------------
udp timeout: 5

OrgId :
-----------------
orgid : 2427270

Resolver config:

RESOLVER IP's
---------------------
```

```
 208.67.220.220
 208.67.222.222
 2620:119:35::35
 2620:119:53::53

Dnscrypt Info:

public_key:
magic_key:
serial number:

ProfileID    DeviceID           Mode          Resolver   Local-Domain     Tag
-------------------------------------------------------------------------------
```

# show radius server-group

To display properties for the RADIUS server group, use the **show radius server-group** command in user EXEC or privileged EXEC mode.

**show radius server-group** {*name* | **all**}

## Syntax Description

| | |
|---|---|
| *name* | Name of the server group. The character string used to name the group of servers must be defined using **the aaa group server radius** command. |
| **all** | Displays properties for all of the server groups. |

## Command Modes

User EXEC (>)

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Use the **show radius server-group** command to display the server groups that you defined by using the **aaa group server radius** command.

The following is sample output from the **show radius server-group all** command:

```
Device# show radius server-group all

Server group radius
    Sharecount = 1  sg_unconfigured = FALSE
    Type = standard  Memlocks = 1
```

This table describes the significant fields shown in the display.

***Table 17: show radius server-group command Field Descriptions***

| Field | Description |
|---|---|
| Server group | Name of the server group. |
| Sharecount | Number of method lists that are sharing this server group. For example, if one method list uses a particular server group, the sharecount would be 1. If two method lists use the same server group, the sharecount would be 2. |
| sg_unconfigured | Server group has been unconfigured. |

| Field | Description |
|---|---|
| Type | The type can be either standard or nonstandard. The type indicates whether the servers in the group accept nonstandard attributes. If all servers within the group are configured with the nonstandard option, the type will be shown as "nonstandard". |
| Memlocks | An internal reference count for the server-group structure that is in memory. The number represents how many internal data structure packets or transactions are holding references to this server group. Memlocks is used internally for memory management purposes. |

# show storm-control

To display broadcast, multicast, or unicast storm control settings on the device or on the specified interface or to display storm-control history, use the **show storm-control** command in user EXEC or privileged EXEC mode.

**show storm-control** [*interface-id*] [**broadcast** | **multicast** | **unicast**]

## Syntax Description

| | |
|---|---|
| *interface-id* | (Optional) Interface ID for the physical port (including type, stack member for stacking-capable devices, module, and port number). |
| **broadcast** | (Optional) Displays broadcast storm threshold setting. |
| **multicast** | (Optional) Displays multicast storm threshold setting. |
| **unicast** | (Optional) Displays unicast storm threshold setting. |

## Command Modes

User EXEC (>)

Privileged EXEC (>)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

When you enter an interface ID, the storm control thresholds appear for the specified interface. If you do not enter an interface ID, settings appear for one traffic type for all ports on the device. If you do not enter a traffic type, settings appear for broadcast storm control.

The following is sample partial output from the **show storm-control** command when no keywords are entered. Because no traffic-type keyword was entered, the broadcast storm control settings appear.

```
Device> show storm-control

Interface Filter State  Upper      Lower     Current
--------- ------------- ---------- --------- ---------
Gi1/0/1   Forwarding    20 pps     10 pps    5 pps
Gi1/0/2   Forwarding    50.00%     40.00%    0.00%
<output truncated>
```

The following is sample output from the **show storm-control** command for a specified interface. Because no traffic-type keyword was entered, the broadcast storm control settings appear.

```
Device> show storm-control gigabitethernet 1/0/1

Interface Filter State  Upper      Lower     Current
--------- ------------- ---------- --------- ---------
Gi1/0/1   Forwarding    20 pps     10 pps    5 pps
```

The following table describes the fields in the show storm-control display:

*Table 18: show storm-control Field Descriptions*

| Field | Description |
|---|---|
| Interface | Displays the ID of the interface. |
| Filter State | Displays the status of the filter:<br>• Blocking—Storm control is enabled, and a storm has occurred.<br>• Forwarding—Storm control is enabled, and no storms have occurred.<br>• Inactive—Storm control is disabled. |
| Upper | Displays the rising suppression level as a percentage of total available bandwidth in packets per second or in bits per second. |
| Lower | Displays the falling suppression level as a percentage of total available bandwidth in packets per second or in bits per second. |
| Current | Displays the bandwidth usage of broadcast traffic or the specified traffic type (broadcast, multicast, or unicast) as a percentage of total available bandwidth. This field is only valid when storm control is enabled. |

# show tech-support acl

To display access control list (ACL)-related information for technical support, use the **show tech-support acl** command in privileged EXEC mode.

**show tech-support acl**

## Syntax Description

This command has no arguments or keywords.

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Gibraltar 16.10.1 | This command was introduced. |
| Cisco IOS XE Gibraltar 16.11.1 | |

## Usage Guidelines

The output of the **show tech-support acl** command is very long. To better manage this output, you can redirect the output to an external file (for example, **show tech-support acl | redirect flash:***show_tech_acl.txt*) in the local writable storage file system or remote file system.

The output of this command displays the following commands:

**Note** On stackable platforms, these commands are executed on every switch in the stack. On modular platforms, like Catalyst 9400 Series Switches, these commands are run only on the active switch.

**Note** The following list of commands is a sample of the commands available in the output; these may differ based on the platform.

- **show clock**
- **show version**
- **show running-config**
- **show module**
- **show interface**
- **show access-lists**
- **show logging**
- **show platform software fed switch** *switch-number* **acl counters hardware**
- **show platform software fed switch** *switch-number* **ifm mapping**
- **show platform hardware fed switch** *switch-number* **fwd-asic drops exceptions**
- **show platform software fed switch** *switch-number* **acl info**

- **show platform software fed switch** *switch-number* **acl**
- **show platform software fed switch** *switch-number* **acl usage**
- **show platform software fed switch** *switch-number* **acl policy intftype all cam**
- **show platform software fed switch** *switch-number* **acl cam brief**
- **show platform software fed switch** *switch-number* **acl policy intftype all vcu**
- **show platform hardware fed switch** *switch-number* **acl resource usage**
- **show platform hardware fed switch** *switch-number* **fwd-asic resource tcam table acl**
- **show platform hardware fed switch** *switch-number* **fwd-asic resource tcam utilization**
- **show platform software fed switch** *switch-number* **acl counters hardware**
- **show platform software classification switch** *switch-number* **all F0 class-group-manager class-group**
- **show platform software process database forwarding-manager switch** *switch-number* **R0 summary**
- **show platform software process database forwarding-manager switch** *switch-number* **F0 summary**
- **show platform software object-manager switch** *switch-number* **F0 pending-ack-update**
- **show platform software object-manager switch** *switch-number* **F0 pending-issue-update**
- **show platform software object-manager switch** *switch-number* **F0 error-object**
- **show platform software peer forwarding-manager switch** *switch-number* **F0**
- **show platform software access-list switch** *switch-number* **f0 statistics**
- **show platform software access-list switch** *switch-number* **r0 statistics**
- **show platform software trace message fed switch** *switch-number*
- **show platform software trace message forwarding-manager switch** *switch-number* **F0**
- **show platform software trace message forwarding-manager switch R0** *switch-number* **R0**

## Examples

The following is sample output from the **show tech-support acl** command:

```
Device# show tech-support acl

.
.
.
------------------ show platform software fed switch 1 acl cam brief ------------------

Printing entries for region ACL_CONTROL (143) type 6 asic 0
============================================================
TAQ-4 Index-0 (A:0,C:0) Valid StartF-1 StartA-1 SkipF-0 SkipA-0
Output IPv4 VACL

 VCU Result: Not In-Use

 L3 Length: 0000, L3 Protocol: 17 (UDP), L3 Tos: 00

 Source Address/Mask
 0.0.0.0/0.0.0.0
```

```
 Destination Address/Mask
 0.0.0.0/0.0.0.0

 Router MAC: Disabled, Not First Fragment: Disabled, Small Offset: Disabled

 L4 Source Port/Mask   L4 Destination Port/Mask
 0x0044 (68)/0xffff     0x0043 (67)/0xffff

 TCP Flags: 0x00 ( NOT SET )

 ACTIONS:  Forward L3, Forward L2, Logging Disabled
 ACL Priority: 2 (15 is Highest Priority)

----------------------------------------
TAQ-4 Index-1 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Output IPv4 VACL

 VCU Result: Not In-Use

 L3 Length: 0000, L3 Protocol: 17 (UDP), L3 Tos: 00

 Source Address/Mask
 0.0.0.0/0.0.0.0
 Destination Address/Mask
 0.0.0.0/0.0.0.0

 Router MAC: Disabled, Not First Fragment: Disabled, Small Offset: Disabled

 L4 Source Port/Mask   L4 Destination Port/Mask
 0x0043 (67)/0xffff     0x0044 (68)/0xffff

 TCP Flags: 0x00 ( NOT SET )

 ACTIONS:  Forward L3, Forward L2, Logging Disabled
 ACL Priority: 2 (15 is Highest Priority)

----------------------------------------
TAQ-4 Index-2 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Output IPv4 VACL

 VCU Result: Not In-Use

 L3 Length: 0000, L3 Protocol: 17 (UDP), L3 Tos: 00

 Source Address/Mask
 0.0.0.0/0.0.0.0
 Destination Address/Mask
 0.0.0.0/0.0.0.0

 Router MAC: Disabled, Not First Fragment: Disabled, Small Offset: Disabled

 L4 Source Port/Mask   L4 Destination Port/Mask
 0x0043 (67)/0xffff     0x0043 (67)/0xffff

 TCP Flags: 0x00 ( NOT SET )

 ACTIONS:  Forward L3, Forward L2, Logging Disabled
 ACL Priority: 2 (15 is Highest Priority)

----------------------------------------
TAQ-4 Index-3 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Input IPv4 PACL

 VCU Result: Not In-Use
```

```
 L3 Length: 0000, L3 Protocol: 00 (HOPOPT), L3 Tos: 00

 Source Address/Mask
 0.0.0.0/0.0.0.0
 Destination Address/Mask
 0.0.0.0/0.0.0.0

 Router MAC: Disabled, Not First Fragment: Disabled, Small Offset: Disabled

 L4 Source Port/Mask   L4 Destination Port/Mask
 0x0000 (0)/0x0000     0x0000 (0)/0x0000

 TCP Flags: 0x00 ( NOT SET )

 ACTIONS:  Drop L3, Drop L2, Logging Disabled
 ACL Priority: 2 (15 is Highest Priority)

----------------------------------------
TAQ-4 Index-4 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Output IPv4 PACL

 VCU Result: Not In-Use

 L3 Length: 0000, L3 Protocol: 00 (HOPOPT), L3 Tos: 00

 Source Address/Mask
 0.0.0.0/0.0.0.0
 Destination Address/Mask
 0.0.0.0/0.0.0.0

 Router MAC: Disabled, Not First Fragment: Disabled, Small Offset: Disabled

 L4 Source Port/Mask   L4 Destination Port/Mask
 0x0000 (0)/0x0000     0x0000 (0)/0x0000

 TCP Flags: 0x00 ( NOT SET )

 ACTIONS:  Drop L3, Drop L2, Logging Disabled
 ACL Priority: 2 (15 is Highest Priority)

----------------------------------------
TAQ-4 Index-5 (A:0,C:0) Valid StartF-0 StartA-0 SkipF-0 SkipA-0
Output MAC PACL

 VLAN ID/MASK : 0x000 (000)/0x000

 Source MAC/Mask : 0000.0000.0000/0000.0000.0000

 Destination MAC/Mask : 0000.0000.0000/0000.0000.0000

 isSnap: Disabled, isLLC: Disabled

 ACTIONS:  Drop L3, Drop L2, Logging Disabled
 ACL Priority: 2 (15 is Highest Priority)

.
.
.
```

Output fields are self-explanatory.

# show tech-support identity

To display identity/802.1x-related information for technical support, use the **show tech-support identity** command in privileged EXEC mode.

**show tech-support identity mac** *mac-address* **interface** *interface-name*

## Syntax Description

| | |
|---|---|
| **mac** *mac-address* | Displays information about the client MAC address. |
| **interface** *interface-name* | Displays information about the client interface. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Gibraltar 16.10.1 | This command was introduced. |
| Cisco IOS XE Gibraltar 16.11.1 | |

## Usage Guidelines

The output of the **show tech-support platform** command is very long. To better manage this output, you can redirect the output to an external file (for example, **show tech-support identity mac** *mac-address* **interface** *interface-name* **| redirect flash:***filename*) in the local writable storage file system or remote file system.

The output of this command displays the following commands:

- **show clock**
- **show module**
- **show version**
- **show switch**
- **show redundancy**
- **show dot1x statistics**
- **show ip access-lists**
- **show interface**
- **show ip interface brief**
- **show vlan brief**
- **show running-config**
- **show logging**
- **show interface controller**
- **show platform authentication sbinfo interface**

- **show platform host-access-table**
- **show platform pm port-data**
- **show spanning-tree interface**
- **show access-session mac detail**
- **show platform authentication session mac**
- **show device-tracking database mac details**
- **show mac address-table address**
- **show access-session event-logging mac**
- **show authentication sessions mac details R0**
- **show ip admission cache R0**
- **show platform software wired-client R0**
- **show platform software wired-client F0**
- **show platform software process database forwarding-manager R0 summary**
- **show platform software process database forwarding-manager F0 summary**
- **show platform software object-manager F0 pending-ack-update**
- **show platform software object-manager F0 pending-issue-update**
- **show platform software object-manager F0 error-object**
- **show platform software peer forwarding-manager R0**
- **show platform software peer forwarding-manager F0**
- **show platform software VP R0 summary**
- **show platform software VP F0 summary**
- **show platform software fed punt cpuq**
- **show platform software fed punt cause summary**
- **show platform software fed inject cause summary**
- **show platform hardware fed fwd-asic drops exceptions**
- **show platform hardware fed fwd-asic resource tcam table acl**
- **show platform software fed acl counter hardware**
- **show platform software fed matm macTable**
- **show platform software fed ifm mappings**
- **show platform software trace message fed reverse**
- **show platform software trace message forwarding-manager R0 reverse**
- **show platform software trace message forwarding-manager F0 reverse**

- **show platform software trace message smd R0 reverse**
- **show authentication sessions mac details**
- **show platform software wired-client**
- **show platform software process database forwarding-manager summary**
- **show platform software object-manager pending-ack-update**
- **show platform software object-manager pending-issue-update**
- **show platform software object-manager error-object**
- **show platform software peer forwarding-manager**
- **show platform software VP summary**
- **show platform software trace message forwarding-manager reverse**
- **show ip admission cache**
- **show platform software trace message smd reverse**
- **show platform software fed punt cpuq**
- **show platform software fed punt cause summary**
- **show platform software fed inject cause summary**
- **show platform hardware fed fwd-asic drops exceptions**
- **show platform hardware fed fwd-asic resource tcam table acl**
- **show platform software fed acl counter hardware**
- **show platform software fed matm macTable**
- **show platform software fed ifm mappings**
- **show platform software trace message fed reverse**

## Examples

The following is sample output from the **show tech-support identity** command:

```
Device# show tech-support identity mac 0000.0001.0003 interface gigabitethernet1/0/1

.
.
.
------------------ show platform software peer forwarding-manager R0  ------------------

IOSD Connection Information:

  MQIPC (reader) Connection State: Connected, Read-selected
    Connections: 1, Failures: 22
    3897 packet received (0 dropped), 466929 bytes
    Read attempts: 2352, Yields: 0
  BIPC Connection state: Connected, Ready
    Accepted: 1, Rejected: 0, Closed: 0, Backpressures: 0
    36 packets sent, 2808 bytes

SMD Connection Information:
```

```
  MQIPC (reader) Connection State: Connected, Read-selected
    Connections: 1, Failures: 30
    0 packet received (0 dropped), 0 bytes
    Read attempts: 1, Yields: 0
  MQIPC (writer) Connection State: Connected, Ready
    Connections: 1, Failures: 0, Backpressures: 0
    0 packet sent, 0 bytes

FP Peers Information:

  Slot: 0
    Peer state: connected
    OM ID: 0, Download attempts: 638
      Complete: 638, Yields: 0, Spurious: 0
      IPC Back-Pressure: 0, IPC-Log Back-Pressure: 0
    Back-Pressure asserted for IPC: 0, IPC-Log: 1
    Number of FP FMAN peer connection expected: 7
    Number of FP FMAN online msg received: 1
    IPC state: unknown

    Config IPC Context:
      State: Connected, Read-selected
      BIPC Handle: 0xdf3d48e8, BIPC FD: 36, Peer Context: 0xdf3e7158
      Tx Packets: 688, Messages: 2392, ACKs: 36
      Rx Packets: 37, Bytes: 2068

      IPC Log:
        Peer name: fman-log-bay0-peer0
        Flags: Recovery-Complete
        Send Seq: 36, Recv Seq: 36, Msgs Sent: 0, Msgs Recovered: 0

    Upstream FMRP IPC Context:
      State: Connected, Read-selected
      BIPC Handle: 0xdf3e7308, BIPC FD: 37, Peer Context: 0xdf3e7158
      TX Packets: 0, Bytes: 0, Drops: 0
      Rx Packets: 0, Bytes: 0

    Upstream FMRP-IOSd IPC Context:
      State: Connected, Read-selected
      BIPC Handle: 0xdf3f9c38, BIPC FD: 38, Peer Context: 0xdf3e7158
      TX Packets: 0, Bytes: 0, Drops: 0
      Rx Packets: 37, Bytes: 2864
      Rx ACK Requests: 1, Tx ACK Responses: 1

    Upstream FMRP-SMD IPC Context:
      State: Connected, Read-selected
      BIPC Handle: 0xdf40c568, BIPC FD: 39, Peer Context: 0xdf3e7158
      TX Packets: 0, Bytes: 0, Drops: 0
      Rx Packets: 0, Bytes: 0
      Rx ACK Requests: 0, Tx ACK Responses: 0

    Upstream FMRP-WNCD_0 IPC Context:
      State: Connected
      BIPC Handle: 0xdf4317c8, BIPC FD: 41, Peer Context: 0xdf3e7158
      TX Packets: 0, Bytes: 0, Drops: 0
      Rx Packets: 0, Bytes: 0
      Rx ACK Requests: 0, Tx ACK Responses: 0

    Upstream FMRP-WNCMGRD IPC Context:
      State: Connected
      BIPC Handle: 0xdf41ee98, BIPC FD: 40, Peer Context: 0xdf3e7158
      TX Packets: 0, Bytes: 0, Drops: 0
      Rx Packets: 0, Bytes: 0
```

```
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-MOBILITYD IPC Context:
    State: Connected
    BIPC Handle: 0xdf4440f8, BIPC FD: 42, Peer Context: 0xdf3e7158
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

Slot: 1
Peer state: connected
  OM ID: 1, Download attempts: 1
    Complete: 1, Yields: 0, Spurious: 0
    IPC Back-Pressure: 0, IPC-Log Back-Pressure: 0
  Back-Pressure asserted for IPC: 0, IPC-Log: 0
  Number of FP FMAN peer connection expected: 7
  Number of FP FMAN online msg received: 1
  IPC state: unknown

  Config IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf45e4d8, BIPC FD: 48, Peer Context: 0xdf470e18
    Tx Packets: 20, Messages: 704, ACKs: 1
    Rx Packets: 2, Bytes: 108

    IPC Log:
      Peer name: fman-log-bay0-peer1
      Flags: Recovery-Complete
      Send Seq: 1, Recv Seq: 1, Msgs Sent: 0, Msgs Recovered: 0

  Upstream FMRP IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf470fc8, BIPC FD: 49, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0

  Upstream FMRP-IOSd IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf4838f8, BIPC FD: 50, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-SMD IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf496228, BIPC FD: 51, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-WNCD_0 IPC Context:
    State: Connected
    BIPC Handle: 0xdf4bb488, BIPC FD: 53, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-WNCMGRD IPC Context:
    State: Connected
    BIPC Handle: 0xdf4a8b58, BIPC FD: 52, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0
```

```
    Upstream FMRP-MOBILITYD IPC Context:
      State: Connected
      BIPC Handle: 0xdf4cddb8, BIPC FD: 54, Peer Context: 0xdf470e18
      TX Packets: 0, Bytes: 0, Drops: 0
      Rx Packets: 0, Bytes: 0
      Rx ACK Requests: 0, Tx ACK Responses: 0



------------------ show platform software peer forwarding-manager R0  ------------------

IOSD Connection Information:

  MQIPC (reader) Connection State: Connected, Read-selected
    Connections: 1, Failures: 22
    3897 packet received (0 dropped), 466929 bytes
    Read attempts: 2352, Yields: 0
  BIPC Connection state: Connected, Ready
    Accepted: 1, Rejected: 0, Closed: 0, Backpressures: 0
    36 packets sent, 2808 bytes

SMD Connection Information:

  MQIPC (reader) Connection State: Connected, Read-selected
    Connections: 1, Failures: 30
    0 packet received (0 dropped), 0 bytes
    Read attempts: 1, Yields: 0
  MQIPC (writer) Connection State: Connected, Ready
    Connections: 1, Failures: 0, Backpressures: 0
    0 packet sent, 0 bytes

FP Peers Information:

  Slot: 0
    Peer state: connected
    OM ID: 0, Download attempts: 638
      Complete: 638, Yields: 0, Spurious: 0
      IPC Back-Pressure: 0, IPC-Log Back-Pressure: 0
    Back-Pressure asserted for IPC: 0, IPC-Log: 1
    Number of FP FMAN peer connection expected: 7
    Number of FP FMAN online msg received: 1
    IPC state: unknown

    Config IPC Context:
      State: Connected, Read-selected
      BIPC Handle: 0xdf3d48e8, BIPC FD: 36, Peer Context: 0xdf3e7158
      Tx Packets: 688, Messages: 2392, ACKs: 36
      Rx Packets: 37, Bytes: 2068

      IPC Log:
        Peer name: fman-log-bay0-peer0
        Flags: Recovery-Complete
        Send Seq: 36, Recv Seq: 36, Msgs Sent: 0, Msgs Recovered: 0

    Upstream FMRP IPC Context:
      State: Connected, Read-selected
      BIPC Handle: 0xdf3e7308, BIPC FD: 37, Peer Context: 0xdf3e7158
      TX Packets: 0, Bytes: 0, Drops: 0
      Rx Packets: 0, Bytes: 0

    Upstream FMRP-IOSd IPC Context:
      State: Connected, Read-selected
      BIPC Handle: 0xdf3f9c38, BIPC FD: 38, Peer Context: 0xdf3e7158
```

```
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 37, Bytes: 2864
    Rx ACK Requests: 1, Tx ACK Responses: 1

  Upstream FMRP-SMD IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf40c568, BIPC FD: 39, Peer Context: 0xdf3e7158
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-WNCD_0 IPC Context:
    State: Connected
    BIPC Handle: 0xdf4317c8, BIPC FD: 41, Peer Context: 0xdf3e7158
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-WNCMGRD IPC Context:
    State: Connected
    BIPC Handle: 0xdf41ee98, BIPC FD: 40, Peer Context: 0xdf3e7158
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-MOBILITYD IPC Context:
    State: Connected
    BIPC Handle: 0xdf4440f8, BIPC FD: 42, Peer Context: 0xdf3e7158
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

Slot: 1
Peer state: connected
  OM ID: 1, Download attempts: 1
    Complete: 1, Yields: 0, Spurious: 0
    IPC Back-Pressure: 0, IPC-Log Back-Pressure: 0
  Back-Pressure asserted for IPC: 0, IPC-Log: 0
  Number of FP FMAN peer connection expected: 7
  Number of FP FMAN online msg received: 1
  IPC state: unknown

  Config IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf45e4d8, BIPC FD: 48, Peer Context: 0xdf470e18
    Tx Packets: 20, Messages: 704, ACKs: 1
    Rx Packets: 2, Bytes: 108

    IPC Log:
      Peer name: fman-log-bay0-peer1
      Flags: Recovery-Complete
      Send Seq: 1, Recv Seq: 1, Msgs Sent: 0, Msgs Recovered: 0

  Upstream FMRP IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf470fc8, BIPC FD: 49, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0

  Upstream FMRP-IOSd IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf4838f8, BIPC FD: 50, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
```

```
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-SMD IPC Context:
    State: Connected, Read-selected
    BIPC Handle: 0xdf496228, BIPC FD: 51, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-WNCD_0 IPC Context:
    State: Connected
    BIPC Handle: 0xdf4bb488, BIPC FD: 53, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-WNCMGRD IPC Context:
    State: Connected
    BIPC Handle: 0xdf4a8b58, BIPC FD: 52, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0

  Upstream FMRP-MOBILITYD IPC Context:
    State: Connected
    BIPC Handle: 0xdf4cddb8, BIPC FD: 54, Peer Context: 0xdf470e18
    TX Packets: 0, Bytes: 0, Drops: 0
    Rx Packets: 0, Bytes: 0
    Rx ACK Requests: 0, Tx ACK Responses: 0



------------------ show platform software VP R0 summary ------------------


Forwarding Manager Vlan Port Information

  Vlan    Intf-ID   Stp-state
  ------------------------------------------------------------------------------
  1       7           Forwarding
  1       9           Forwarding
  1       17          Forwarding
  1       27          Forwarding
  1       28          Forwarding
  1       29          Forwarding
  1       30          Forwarding
  1       31          Forwarding
  1       40          Forwarding
  1       41          Forwarding


Forwarding Manager Vlan Port Information

  Vlan    Intf-ID   Stp-state
  ------------------------------------------------------------------------------
  1       49          Forwarding
  1       51          Forwarding
  1       63          Forwarding
  1       72          Forwarding
  1       73          Forwarding
  1       74          Forwarding
```

```
------------------ show platform software VP R0 summary ------------------


Forwarding Manager Vlan Port Information

  Vlan    Intf-ID   Stp-state
  ------------------------------------------------------------------------------
  1       7           Forwarding
  1       9           Forwarding
  1       17          Forwarding
  1       27          Forwarding
  1       28          Forwarding
  1       29          Forwarding
  1       30          Forwarding
  1       31          Forwarding
  1       40          Forwarding
  1       41          Forwarding


Forwarding Manager Vlan Port Information

  Vlan    Intf-ID   Stp-state
  ------------------------------------------------------------------------------
  1       49          Forwarding
  1       51          Forwarding
  1       63          Forwarding
  1       72          Forwarding
  1       73          Forwarding
  1       74          Forwarding
.
.
.
```

# show umbrella

To display the Cisco Umbrella Integration feature-related configuration, use the **show umbrella** command in user EXEC or privileged EXEC mode.

**show umbrella** {**config** | **deviceid** [**detailed**] | **dnscrypt**}

## Syntax Description

| | |
|---|---|
| **config** | Displays global configurations of the device. |
| **deviceid** | Displays device registration details. |
| **dnscrypt** | Displays DNSCrypt-related configurations. |

## Command Modes

User EXEC (>)

Privileged EXEC (>)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Examples

The following is a sample output of the **show umbrella config** command:

```
Device> show umbrella config

Umbrella Configuration
=========================
   Token: 0C6ED7E376DD4D2E04492CE7EDFF1A7C00250986
   API-KEY: NONE
   OrganizationID: 2427270
   Local Domain Regex parameter-map name: NONE
   DNSCrypt: Enabled
  Public-key: B735:1140:206F:225D:3E2B:D822:D7FD:691E:A1C3:3CC8:D666:8D0C:BE04:BFAB:CA43:FB79

   UDP Timeout: 5 seconds
   Resolver address:
       1. 208.67.220.220
       2. 208.67.222.222
       3. 2620:119:53::53
       4. 2620:119:35::35
Umbrella Interface Config:
       Number of interfaces with "umbrella out" config: 1
         1. GigabitEthernet1/0/48
             Mode     :  OUT
             VRF      : global(Id: 0)
       Number of interfaces with "umbrella in" config: 1
         1. GigabitEthernet1/0/1
             Mode       : IN
             DCA        : Disabled
             Tag        : test
             Device-id  : 010a2c41b8ab019c
             VRF        : global(Id: 0)

   Configured Umbrella Parameter-maps:
```

```
        1. global
```

The following is a sample output of the **show umbrella deviceid detailed** command:

```
Device> show umbrella deviceid detailed

Device registration details
 1.GigabitEthernet1/0/2
      Tag               : guest
      Device-id         : 010a6aef0b443f0f
      Description       : Device Id received successfully
      WAN interface     : GigabitEthernet1/0/1
      WAN VRF used      : global(Id: 0)
```

The following is a sample output of the **show umbrella dnscrypt** command:

```
Device> show umbrella dnscrypt

DNSCrypt: Enabled
Public-key: B735:1140:206F:225D:3E2B:D822:D7FD:691E:A1C3:3CC8:D666:8D0C:BE04:BFAB:CA43:FB79
Certificate Update Status:
Last Successful Attempt : 10:55:40 UTC Apr 14 2016
Last Failed Attempt : 10:55:10 UTC Apr 14 2016
Certificate Details:
Certificate Magic : DNSC
Major Version : 0x0001
Minor Version : 0x0000
Query Magic : 0x717744506545635A
Serial Number : 1435874751
Start Time : 1435874751 (22:05:51 UTC Jul 2 2015)
End Time : 1467410751 (22:05:51 UTC Jul 1 2016)
Server Public Key :
ABA1:F000:D394:8045:672D:73E0:EAE6:F181:19D0:2A62:3791:EFAD:B04E:40B7:B6F9:C40B
Client Secret Key Hash :
BBC3:409F:5CB5:C3F3:06BD:A385:78DA:4CED:62BC:3985:1C41:BCCE:1342:DF13:B71E:F4CF
Client Public key :
ECE2:8295:2157:6797:6BE2:C563:A5A9:C5FC:C20D:ADAF:EB3C:A1A2:C09A:40AD:CAEA:FF76
NM key Hash :
F9C2:2C2C:330A:1972:D484:4DD8:8E5C:71FF:6775:53A7:0344:5484:B78D:01B1:B938:E884
```

# show vlan access-map

To display information about a particular VLAN access map or for all VLAN access maps, use the **show vlan access-map** command in privileged EXEC mode.

**show vlan access-map** [*map-name*]

## Syntax Description

| | |
|---|---|
| *map-name* | (Optional) Name of a specific VLAN access map. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Examples

The following is sample output from the **show vlan access-map** command:

```
Device# show vlan access-map

Vlan access-map "vmap4"  10
  Match clauses:
    ip  address: al2
  Action:
    forward
Vlan access-map "vmap4"  20
  Match clauses:
    ip  address: al2
  Action:
    forward
```

# show vlan filter

To display information about all VLAN filters or about a particular VLAN or VLAN access map, use the **show vlan filter** command in privileged EXEC mode.

**show vlan filter** {**access-map** *name* | **vlan** *vlan-id*}

## Syntax Description

| | |
|---|---|
| **access-map** *name* | (Optional) Displays filtering information for the specified VLAN access map. |
| **vlan** *vlan-id* | (Optional) Displays filtering information for the specified VLAN. The range is 1 to 4094. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Examples

The following is sample output from the **show vlan filter** command:

```
Device# show vlan filter

VLAN Map map_1 is filtering VLANs:
  20-22
```

# show vlan group

To display the VLANs that are mapped to VLAN groups, use the **show vlan group** command in privileged EXEC mode.

**show vlan group** [**group-name** *vlan-group-name* [**user_count**]]

## Syntax Description

| | |
|---|---|
| **group-name** *vlan-group-name* | (Optional) Displays the VLANs mapped to the specified VLAN group. |
| **user_count** | (Optional) Displays the number of users in each VLAN mapped to a specified VLAN group. |

## Command Modes

Privileged EXEC (#)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **show vlan group** command displays the existing VLAN groups and lists the VLANs and VLAN ranges that are members of each VLAN group. If you enter the **group-name** keyword, only the members of the specified VLAN group are displayed.

## Examples

This example shows how to display the members of a specified VLAN group:

```
Device# show vlan group group-name group2
vlan group group1 :40-45
```

This example shows how to display number of users in each of the VLANs in a group:

```
Device# show vlan group group-name group2 user_count

  VLAN     : Count
--------------------
  40         : 5
  41         : 8
  42         : 12
  43         : 2
  44         : 9
  45         : 0
```

# ssci-based-on-sci

To compute the Short Secure Channel Identifier (SSCI) value based on the Secure Channel Identifier (SCI) value, use the **ssci-based-on-sci** command in MKA-policy configuration mode. To disable SSCI computation based on SCI, use the **no** form of this command.

**ssci-based-on-sci**
**no ssci-based-on-sci**

## Syntax Description

This command has no arguments or keywords.

## Command Default

SSCI value computation based on SCI value is disabled.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Gibraltar 16.12.3 | This command was introduced. |

## Usage Guidelines

The higher the SCI value, the lower is the SSCI value.

## Examples

The following example shows how to enable the SSCI computation based on SCI:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# ssci-based-on-sci
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **confidentiality-offset** | Sets the confidentiality offset for MACsec operations. |
| **delay-protection** | Configures MKA to use delay protection in sending MKPDU. |
| **include-icv-indicator** | Includes ICV indicator in MKPDU. |
| **key-server** | Configures MKA key-server options. |
| **macsec-cipher-suite** | Configures cipher suite for deriving SAK. |
| **sak-rekey** | Configures the SAK rekey interval. |
| **send-secure-announcements** | Configures MKA to send secure announcements in sending MKPDUs. |
| **use-updated-eth-header** | Uses the updated Ethernet header for ICV calculation. |

# storm-control

To enable broadcast, multicast, or unicast storm control and to set threshold levels on an interface, use the **storm-control** command in interface configuration mode. To return to the default setting, use the **no** form of this command.

**storm-control** {**action** {**shutdown** | **trap**} | {**broadcast** | **multicast** | **unicast** | **unknown-unicast**} **level** {*level* [*level-low*] | **bps** *bps* [*bps-low*] | **pps** *pps* [*pps-low*]}}
**no storm-control** {**action** {**shutdown** | **trap**} | {**broadcast** | **multicast** | **unicast** | **unknown-unicast**} **level**}

## Syntax Description

| | |
|---|---|
| **action** | Specifies the action taken when a storm occurs on a port. The default action is to filter traffic and to not send an Simple Network Management Protocol (SNMP) trap. |
| **shutdown** | Disables the port during a storm. |
| **trap** | Sends an SNMP trap when a storm occurs. |
| **broadcast** | Enables broadcast storm control on the interface. |
| **multicast** | Enables multicast storm control on the interface. |
| **unicast** | Enables unicast storm control on the interface. |
| **unknown-unicast** | Enables unknown unicast storm control on an interface. |
| **level** | Specifies the rising and falling suppression levels as a percentage of total bandwidth of the port. |
| *level* | Rising suppression level, up to two decimal places. The range is 0.00 to 100.00. Block the flooding of storm packets when the value specified for level is reached. |
| *level-low* | (Optional) Falling suppression level, up to two decimal places. The range is 0.00 to 100.00. This value must be less than or equal to the rising suppression value. If you do not configure a falling suppression level, it is set to the rising suppression level. |
| **level bps** | Specifies the rising and falling suppression levels as a rate in bits per second at which traffic is received on the port. |
| *bps* | Rising suppression level, up to 1 decimal place. The range is 0.0 to 10000000000.0. Block the flooding of storm packets when the value specified for bps is reached.<br>You can use metric suffixes such as k, m, and g for large number thresholds. |
| *bps-low* | (Optional) Falling suppression level, up to 1 decimal place. The range is 0.0 to 10000000000.0. This value must be equal to or less than the rising suppression value.<br>You can use metric suffixes such as k, m, and g for large number thresholds. |
| **level pps** | Specifies the rising and falling suppression levels as a rate in packets per second at which traffic is received on the port. |

| | |
|---|---|
| *pps* | Rising suppression level, up to 1 decimal place. The range is 0.0 to 10000000000.0. Block the flooding of storm packets when the value specified for pps is reached.<br>You can use metric suffixes such as k, m, and g for large number thresholds. |
| *pps-low* | (Optional) Falling suppression level, up to 1 decimal place. The range is 0.0 to 10000000000.0. This value must be equal to or less than the rising suppression value.<br>You can use metric suffixes such as k, m, and g for large number thresholds. |

**Command Default**

Broadcast, multicast, and unicast storm control are disabled.

The default action is to filter traffic and to not send an SNMP trap.

**Command Modes**

Interface configuration (config-if)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |
| Cisco IOS XE Gibraltar 16.11.1 | This command was modified. The **unknown-unicast** keyword was added. |

**Usage Guidelines**

The storm-control suppression level can be entered as a percentage of total bandwidth of the port, as a rate in packets per second at which traffic is received, or as a rate in bits per second at which traffic is received.

When specified as a percentage of total bandwidth, a suppression value of 100 percent means that no limit is placed on the specified traffic type. A value of **level 0 0** means that all broadcast, multicast, or unicast traffic on that port is blocked. Storm control is enabled only when the rising suppression level is less than 100 percent. If no other storm-control configuration is specified, the default action is to filter the traffic causing the storm and to send no SNMP traps.

**Note** When the storm control threshold for multicast traffic is reached, all multicast traffic except control traffic, such as bridge protocol data unit (BDPU) and Cisco Discovery Protocol (CDP) frames, are blocked. However, the device does not differentiate between routing updates, such as Open Shortest Path First (OSPF) and regular multicast data traffic, so both types of traffic are blocked.

The **trap** and **shutdown** options are independent of each other.

If you configure the action to be taken as shutdown (the port is error-disabled during a storm) when a packet storm is detected, you must use the **no shutdown** interface configuration command to bring the interface out of this state. If you do not specify the **shutdown** action, specify the action as **trap** (the device generates a trap when a storm is detected).

When a storm occurs and the action is to filter traffic, if the falling suppression level is not specified, the device blocks all traffic until the traffic rate drops below the rising suppression level. If the falling suppression level is specified, the device blocks traffic until the traffic rate drops below this level.

**Note** Storm control is supported on physical interfaces. You can also configure storm control on an EtherChannel. When storm control is configured on an EtherChannel, the storm control settings propagate to the EtherChannel physical interfaces.

When a broadcast storm occurs and the action is to filter traffic, the device blocks only broadcast traffic.

For more information, see the software configuration guide for this release.

This example shows how to enable broadcast storm control with a 75.5-percent rising suppression level:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# storm-control broadcast level 75.5
Device(config-if)# end
```

This example shows how to enable unicast storm control on a port with a 87-percent rising suppression level and a 65-percent falling suppression level:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# storm-control unicast level 87 65
Device(config-if)# end
```

This example shows how to enable multicast storm control on a port with a 2000-packets-per-second rising suppression level and a 1000-packets-per-second falling suppression level:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# storm-control multicast level pps 2k 1k
Device(config-if)# end
```

This example shows how to enable the **shutdown** action on a port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 1/0/1
Device(config-if)# storm-control action shutdown
Device(config-if)# end
```

You can verify your settings by entering the **show storm-control** command.

# switchport port-security aging

To set the aging time and type for secure address entries or to change the aging behavior for secure addresses on a particular port, use the **switchport port-security aging** command in interface configuration mode. To disable port security aging or to set the parameters to their default states, use the **no** form of this command.

**switchport port-security aging** {**static** | **time** *time* | **type** {**absolute** | **inactivity**}}
**no switchport port-security aging** {**static** | **time** | **type**}

## Syntax Description

| | |
|---|---|
| **static** | Enables aging for statically configured secure addresses on this port. |
| **time** *time* | Specifies the aging time for this port. The range is 0 to 1440 minutes. If the time is 0, aging is disabled for this port. |
| **type** | Sets the aging type. |
| **absolute** | Sets absolute aging type. All the secure addresses on this port age out exactly after the time (minutes) specified and are removed from the secure address list. |
| **inactivity** | Sets the inactivity aging type. The secure addresses on this port age out only if there is no data traffic from the secure source address for the specified time period. |

## Command Default

The port security aging feature is disabled. The default time is 0 minutes.

The default aging type is absolute.

The default static aging behavior is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

To enable secure address aging for a particular port, set the aging time to a value other than 0 for that port.

To allow limited time access to particular secure addresses, set the aging type as **absolute**. When the aging time lapses, the secure addresses are deleted.

To allow continuous access to a limited number of secure addresses, set the aging type as **inactivity**. This removes the secure address when it become inactive, and other addresses can become secure.

To allow unlimited access to a secure address, configure it as a secure address, and disable aging for the statically configured secure address by using the **no switchport port-security aging static** interface configuration command.

This example sets the aging time as 2 hours for absolute aging for all the secure addresses on the port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# switchport port-security aging time 120
```

```
Device(config-if)# end
```

This example sets the aging time as 2 minutes for inactivity aging type with aging enabled for configured secure addresses on the port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# switchport port-security aging time 2
Device(config-if)# switchport port-security aging type inactivity
Device(config-if)# switchport port-security aging static
Device(config-if)# end
```

This example shows how to disable aging for configured secure addresses:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet1/0/2
Device(config-if)# no switchport port-security aging static
Device(config-if)# end
```

# switchport port-security mac-address

To configure secure MAC addresses or sticky MAC address learning, use the **switchport port-security mac-address** interface configuration command. To return to the default setting, use the **no** form of this command.

**switchport port-security mac-address** {*mac-address* [**vlan** {*vlan-id* {**access** | **voice**}}] | **sticky** [*mac-address* | **vlan** {*vlan-id* {**access** | **voice**}}]}
**no switchport port-security mac-address** {*mac-address* [**vlan** {*vlan-id* {**access** | **voice**}}] | **sticky** [*mac-address* | **vlan** {*vlan-id* {**access** | **voice**}}]}

## Syntax Description

| | |
|---|---|
| *mac-address* | A secure MAC address for the interface by entering a 48-bit MAC address. You can add additional secure MAC addresses up to the maximum value configured. |
| **vlan** *vlan-id* | (Optional) On a trunk port only, specifies the VLAN ID and the MAC address. If no VLAN ID is specified, the native VLAN is used. |
| **vlan access** | (Optional) On an access port only, specifies the VLAN as an access VLAN. |
| **vlan voice** | (Optional) On an access port only, specifies the VLAN as a voice VLAN.<br>**Note**<br>The **voice** keyword is available only if voice VLAN is configured on a port and if that port is not the access VLAN. |
| **sticky** | Enables the interface for sticky learning. When sticky learning is enabled, the interface adds all secure MAC addresses that are dynamically learned to the running configuration and converts these addresses to sticky secure MAC addresses. |
| *mac-address* | (Optional) A MAC address to specify a sticky secure MAC address. |

## Command Default

No secure MAC addresses are configured.

Sticky learning is disabled.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

A secure port has the following limitations:

- A secure port can be an access port or a trunk port; it cannot be a dynamic access port.
- A secure port cannot be a routed port.
- A secure port cannot be a protected port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
- A secure port cannot belong to a Gigabit or 10-Gigabit EtherChannel port group.

- You cannot configure static secure or sticky secure MAC addresses in the voice VLAN.
- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the Cisco IP phone.
- Voice VLAN is supported only on access ports and not on trunk ports.

Sticky secure MAC addresses have these characteristics:

- When you enable sticky learning on an interface by using the **switchport port-security mac-address sticky** interface configuration command, the interface converts all the dynamic secure MAC addresses, including those that were dynamically learned before sticky learning was enabled, to sticky secure MAC addresses and adds all sticky secure MAC addresses to the running configuration.
- If you disable sticky learning by using the **no switchport port-security mac-address sticky** interface configuration command or the running configuration is removed, the sticky secure MAC addresses remain part of the running configuration but are removed from the address table. The addresses that were removed can be dynamically reconfigured and added to the address table as dynamic addresses.
- When you configure sticky secure MAC addresses by using the **switchport port-security mac-address sticky** *mac-address* interface configuration command, these addresses are added to the address table and the running configuration. If port security is disabled, the sticky secure MAC addresses remain in the running configuration.
- If you save the sticky secure MAC addresses in the configuration file, when the device restarts or the interface shuts down, the interface does not need to relearn these addresses. If you do not save the sticky secure addresses, they are lost. If sticky learning is disabled, the sticky secure MAC addresses are converted to dynamic secure addresses and are removed from the running configuration.
- If you disable sticky learning and enter the **switchport port-security mac-address sticky** *mac-address* interface configuration command, an error message appears, and the sticky secure MAC address is not added to the running configuration.

You can verify your settings by using the **show port-security** command.

This example shows how to configure a secure MAC address and a VLAN ID on a port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/2
Device(config-if)# switchport mode trunk
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security mac-address 1000.2000.3000 vlan 3
Device(config-if)# end
```

This example shows how to enable sticky learning and to enter two sticky secure MAC addresses on a port:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/2
Device(config-if)# switchport port-security mac-address sticky
Device(config-if)# switchport port-security mac-address sticky 0000.0000.4141
```

```
Device(config-if)# switchport port-security mac-address sticky 0000.0000.000f
Device(config-if)# end
```

# switchport port-security maximum

To configure the maximum number of secure MAC addresses, use the **switchport port-security maximum** command in interface configuration mode. To return to the default settings, use the **no** form of this command.

**switchport port-security maximum** *value* [**vlan** [*vlan-list* | [**access** | **voice**]]]
**no switchport port-security maximum** *value* [**vlan** [*vlan-list* | [**access** | **voice**]]]

## Syntax Description

| | |
|---|---|
| *value* | Sets the maximum number of secure MAC addresses for the interface.<br>The default setting is 1. |
| **vlan** | (Optional) For trunk ports, sets the maximum number of secure MAC addresses on a VLAN or range of VLANs. If the **vlan** keyword is not entered, the default value is used. |
| *vlan-list* | (Optional) Range of VLANs separated by a hyphen or a series of VLANs separated by commas. For nonspecified VLANs, the per-VLAN maximum value is used. |
| **access** | (Optional) On an access port only, specifies the VLAN as an access VLAN. |
| **voice** | (Optional) On an access port only, specifies the VLAN as a voice VLAN.<br>**Note**<br>The **voice** keyword is available only if voice VLAN is configured on a port and if that port is not the access VLAN. |

## Command Default

When port security is enabled and no keywords are entered, the default maximum number of secure MAC addresses is 1.

## Command Modes

Interface configuration (config-if

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The maximum number of secure MAC addresses that you can configure on a device is set by the maximum number of available MAC addresses allowed in the system. This number is determined by the active Switch Database Management (SDM) template. See the **sdm prefer** command. This number represents the total of available MAC addresses, including those used for other Layer 2 functions and any other secure MAC addresses configured on interfaces.

A secure port has the following limitations:

- A secure port can be an access port or a trunk port; it cannot be a dynamic access port.
- A secure port cannot be a routed port.
- A secure port cannot be a protected port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).

- A secure port cannot belong to a Fast EtherChannel or Gigabit EtherChannel or 10-Gigabit EtherChannel port group.
- When you enable port security on an interface that is also configured with a voice VLAN, set the maximum allowed secure addresses on the port to two. When the port is connected to a Cisco IP phone, the IP phone requires one MAC address. The Cisco IP phone address is learned on the voice VLAN, but is not learned on the access VLAN. If you connect a single PC to the Cisco IP phone, no additional MAC addresses are required. If you connect more than one PC to the Cisco IP phone, you must configure enough secure addresses to allow one for each PC and one for the Cisco IP phone.

  Voice VLAN is supported only on access ports and not on trunk ports.
- When you enter a maximum secure address value for an interface, if the new value is greater than the previous value, the new value overrides the previously configured value. If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.

  Setting a maximum number of addresses to one and configuring the MAC address of an attached device ensures that the device has the full bandwidth of the port.

When you enter a maximum secure address value for an interface, this occurs:

- If the new value is greater than the previous value, the new value overrides the previously configured value.
- If the new value is less than the previous value and the number of configured secure addresses on the interface exceeds the new value, the command is rejected.

You can verify your settings by using the **show port-security** command.

This example shows how to enable port security on a port and to set the maximum number of secure addresses to 5. The violation mode is the default, and no secure MAC addresses are configured.

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet 2/0/2
Device(config-if)# switchport mode access
Device(config-if)# switchport port-security
Device(config-if)# switchport port-security maximum 5
Device(config-if)# end
```

# switchport port-security violation

To configure secure MAC address violation mode or the action to be taken if port security is violated, use the **switchport port-security violation** command in interface configuration mode. To return to the default settings, use the **no** form of this command.

**switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**}
**no switchport port-security violation** {**protect** | **restrict** | **shutdown** | **shutdown vlan**}

## Syntax Description

| | |
|---|---|
| **protect** | Sets the security violation protect mode. |
| **restrict** | Sets the security violation restrict mode. |
| **shutdown** | Sets the security violation shutdown mode. |
| **shutdown vlan** | Sets the security violation mode to per-VLAN shutdown. |

## Command Default

The default violation mode is **shutdown**.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

In the security violation protect mode, when the number of port secure MAC addresses reaches the maximum limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses to drop below the maximum value or increase the number of maximum allowable addresses. You are not notified that a security violation has occurred.

**Note** We do not recommend configuring the protect mode on a trunk port. The protect mode disables learning when any VLAN reaches its maximum limit, even if the port has not reached its maximum limit.

In the security violation restrict mode, when the number of secure MAC addresses reaches the limit allowed on the port, packets with unknown source addresses are dropped until you remove a sufficient number of secure MAC addresses or increase the number of maximum allowable addresses. An SNMP trap is sent, a syslog message is logged, and the violation counter increments.

In the security violation shutdown mode, the interface is error-disabled when a violation occurs and the port LED turns off. An SNMP trap is sent, a syslog message is logged, and the violation counter increments. When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause psecure-violation** global configuration command, or you can manually re-enable it by entering the **shutdown** and **no shutdown** interface configuration commands.

When the security violation mode is set to per-VLAN shutdown, only the VLAN on which the violation occurred is error-disabled.

A secure port has the following limitations:

- A secure port can be an access port or a trunk port; it cannot be a dynamic access port.
- A secure port cannot be a routed port.
- A secure port cannot be a protected port.
- A secure port cannot be a destination port for Switched Port Analyzer (SPAN).
- A secure port cannot belong to a Fast EtherChannel or Gigabit EtherChannel or 10-Gigabit EtherChannel port group.

  A security violation occurs when the maximum number of secure MAC addresses are in the address table and a station whose MAC address is not in the address table attempts to access the interface or when a station whose MAC address is configured as a secure MAC address on another secure port attempts to access the interface.

  When a secure port is in the error-disabled state, you can bring it out of this state by entering the **errdisable recovery cause** *psecure-violation* global configuration command. You can manually re-enable the port by entering the **shutdown** and **no shutdown** interface configuration commands or by using the **clear errdisable interface** privileged EXEC command.

You can verify your settings by using the **show port-security** privileged EXEC command.

This example shows how to configure a port to shut down only the VLAN if a MAC security violation occurs:

```
Device> enable
Device# configure terminal
Device(config)# interface gigabitethernet2/0/2
Device(config)# switchport port-security violation shutdown vlan
Device(config)# exit
```

# tacacs server

To configure the TACACS+ server for IPv6 or IPv4 and enter TACACS+ server configuration mode, use the **tacacs server** command in global configuration mode. To remove the configuration, use the **no** form of this command.

**tacacs server** *name*
**no tacacs server**

## Syntax Description

| | |
|---|---|
| *name* | Name of the private TACACS+ server host. |

## Command Default

No TACACS+ server is configured.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **tacacs server** command configures the TACACS server using the *name* argument and enters TACACS+ server configuration mode. The configuration is applied once you have finished configuration and exited TACACS+ server configuration mode.

## Examples

The following example shows how to configure the TACACS server using the name server1 and enter TACACS+ server configuration mode to perform further configuration:

```
Device> enable
Device# configure terminal
Device(config)# tacacs server server1
Device(config-server-tacacs)# end
```

## Related Commands

| Command | Description |
|---|---|
| **address ipv6 (TACACS+)** | Configures the IPv6 address of the TACACS+ server. |
| **key (TACACS+)** | Configures the per-server encryption key on the TACACS+ server. |
| **port (TACACS+)** | Specifies the TCP port to be used for TACACS+ connections. |
| **send-nat-address (TACACS+)** | Sends a client's post-NAT address to the TACACS+ server. |
| **single-connection (TACACS+)** | Enables all TACACS packets to be sent to the same server using a single TCP connection. |
| **timeout(TACACS+)** | Configures the time to wait for a reply from the specified TACACS server. |

# tls

To configure Transport Layer Security (TLS) parameters, use the **tls** command in radius server configuration mode. To return to the default setting, use the **no** form of this command.

**tls** [ **connectiontimeout** *connection-timeout-value* | **idletimeout** *idle-timeout-value* | [ **ip** | **ipv6** ] { **radius source-interface** *interface-name* | **vrf forwarding** *forwarding-table-name* } | **match-server-identity** { **email-address** *email-address* | **hostname** *hostname* | **ip-address** *ip-address* } | **port** *port-number* | **retries** *number-of-connection-retries* | **trustpoint** { **client** *trustpoint name* | **server** *trustpoint name* } | **watchdoginterval** *interval* ]

**no tls**

## Syntax Description

| | |
|---|---|
| **connectiontimeout** *connection-timeout-value* | (Optional) Configures the DTLS connection timeout value. |
| **idletimeout** *idle-timeout-value* | (Optional) Configures the DTLS idle timeout value. |
| [**ip** \| **ipv6**] {**radius source-interface** *interface-name* \| **vrf forwarding** *forwarding-table-name*} | (Optional) Configures IP or IPv6 source parameters. |
| **match-server-identity** {**email-address** *email-address* \| **hostname** *host-name* \| **ip-address** *ip-address*} | Configures RadSec certification validation parameters. |
| **port** *port-number* | (Optional) Configures the DTLS port number. |
| **retries** *number-of-connection-retries* | (Optional) Configures the number of DTLS connection retries. |
| **trustpoint** {**client** *trustpoint name* \| **server** *trustpoint name*} | (Optional) Configures the DTLS trustpoint for the client and the server. |
| **watchdoginterval** *interval* | (Optional) Configures the watchdog interval. This enables CoA requests to be received on the same authentication channel. It also serves as keepalive to keep the TLS tunnel up, and re-establishes the tunnel if it is torn down.<br>**Note**<br>**watchdoginterval** value must be lesser than **idletimeout** for the established tunnel to remain up. |

## Command Default

- The default value of TLS connection timeout is 5 seconds.
- The default value of TLS idle timeout is 60 seconds.
- The default TLS port number is 2083.
- The default value of TLS connection retries is 5.
- The default value of watchdog interval is 0.

**Command Modes**

Radius server configuration mode (config-radius-server)

**Command History**

| Release | Modification |
|---|---|
| Cisco IOS XE Fuji 16.9.1 | This command was introduced. |
| Cisco IOS XE Gibraltar 16.10.1 | The **match-server-identity** keyword was introduced. |
| Cisco IOS XE Amsterdam 17.1.1 | The **ipv6** keyword was introduced. |
| Cisco IOS XE Bengaluru 17.6.1 | The **watchdoginterval** keyword was introduced. |

**Usage Guidelines**

We recommended that you use the same server type, either only TLS or only Datagram Transport Layer Security (DTLS), under a authentication, authorization, and accounting (AAA) server group.

**Examples**

The following example shows how to configure the TLS idle timeout value to 5 seconds:

```
Device> enable
Device# configure terminal
Device(config)# radius server R1
Device(config-radius-server)# tls idletimeout 5
Device(config-radius-server)# end
```

**Related Commands**

| Command | Description |
|---|---|
| **show aaa servers** | Displays information related to the TLS server. |
| **clear aaa counters servers radius** | Clears the RADIUS TLS-specific statistics. |
| **debug radius radsec** | Enables RADIUS TLS-specific debugs. |

# tlv (mDNS)

To configure the list of Type Length Value (TLVs) in mDNS protocol configuration mode, use the **tlv** command. Use the **no** form of the command to remove indvidial TLVs from the mDNS protocol TLV filter.

**tlv** { **name** { **device-version** | **pointer-records** | **srv-records** | **text-records** } | **number** *tlv-number* }
**no tlv** { **name** { **device-version** | **pointer-records** | **srv-records** | **text-records** } | **number** *tlv-number* }

## Syntax Description

| | |
|---|---|
| **name** | Specifies the TLV type name to be added to the mDNS protocol filter list. |
| **device-version** | Specifies the mDNS device version. |
| **pointer-records** | Specifies the mDNS PTR records. |
| **srv-records** | Specifies the mDNS SRV records. |
| **text-records** | Specifies the mDNS text records. |
| **number** | Specifies the TLV type number to be added to the mDNS protocol filter list. |
| *tlv-number* | TLV type number to be added. Range is from 1 to 16. |

## Command Default

TLV list is not configured.

## Command Modes

mDNS protocol configuration mode (config-sensor-mdnslist)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE 17.14.1 | This command was introduced. |

## Usage Guidelines

Use the **tlv** command to configure the list of TLVs in mDNS protocol configuration mode. The **srv-records** option provides the device hostname to the user.

### Example

The following examples shows how to configure a list of TLVs in mDNS protocol configuration:

```
Device> enable
Device# configure terminal
Device(config)# device-sensor filter-list mdns list mdns-list
Device(config-sensor-mdnslist)# tlv name srv-records
```

# token (Parameter Map)

To configure the application programming interface (API) token used for authorization during device registration, use the **token** command in parameter-map type inspect configuration mode. To remove the unique identifier, use the **no** form of this command.

**token** *value*
**no token**

## Syntax Description

| | |
|---|---|
| *value* | The API token. You can obtain this from the Cisco Umbrella registration server. |

## Command Default

No token is created for the parameter map.

## Command Modes

Parameter-map type inspect configuration (config-profile)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Usage Guidelines

The **token** command is a mandatory configuration for the **umbrella in** and **umbrella out** commands.

To change an existing token to a new token, remove the **umbrella in** command and reconfigure it on the interface for policies of the new token to be applied.

## Examples

The following example shows how to configure a Cisco Umbrella token:

```
Device> enable
Device# configure terminal
Device(config)# parameter-map type umbrella global
Device(config-profile)# token AADDD5FF6E510B28921A20C9B98EEEFF
```

## Related Commands

| Command | Description |
|---|---|
| **parameter-map type umbrella global** | Configures a parameter-map type in umbrella mode. |
| **umbrella** | Configures the Cisco Umbrella Connector on an interface. |

# tracking (IPv6 snooping)

To override the default tracking policy on a port, use the **tracking** command in IPv6 snooping policy configuration mode.

**tracking** {**enable** [**reachable-lifetime** {*value* | **infinite**}] | **disable** [**stale-lifetime** {*value* | **infinite**}

## Syntax Description

| | |
|---|---|
| **enable** | Enables tracking. |
| **reachable-lifetime** | (Optional) Specifies the maximum amount of time a reachable entry is considered to be directly or indirectly reachable without proof of reachability.<br>• The **reachable-lifetime** keyword can be used only with the **enable** keyword.<br>• Use of the **reachable-lifetime** keyword overrides the global reachable lifetime configured by the **ipv6 neighbor binding reachable-lifetime** command. |
| *value* | Lifetime value, in seconds. The range is from 1 to 86400, and the default is 300. |
| **infinite** | Keeps an entry in a reachable or stale state for an infinite amount of time. |
| **disable** | Disables tracking. |
| **stale-lifetime** | (Optional) Keeps the time entry in a stale state, which overwrites the global stale-lifetime configuration.<br>• The stale lifetime is 86,400 seconds.<br>• The **stale-lifetime** keyword can be used only with the **disable** keyword.<br>• Use of the **stale-lifetime** keyword overrides the global stale lifetime configured by the **ipv6 neighbor binding stale-lifetime** command. |

## Command Default

The time entry is kept in a reachable state.

## Command Modes

IPv6 snooping configuration (config-ipv6-snooping)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

The **tracking** command overrides the default tracking policy set by the **ipv6 neighbor tracking** command on the port on which this policy applies. This function is useful on trusted ports where, for example, you may not want to track entries but want an entry to stay in the binding table to prevent it from being stolen.

The **reachable-lifetime** keyword is the maximum time an entry will be considered reachable without proof of reachability, either directly through tracking or indirectly through IPv6 snooping. After the **reachable-lifetime** value is reached, the entry is moved to stale. Use of the **reachable-lifetime** keyword with the tracking command overrides the global reachable lifetime configured by the **ipv6 neighbor binding reachable-lifetime** command.

The **stale-lifetime** keyword is the maximum time an entry is kept in the table before it is deleted or the entry is proven to be reachable, either directly or indirectly. Use of the **reachable-lifetime** keyword with the **tracking** command overrides the global stale lifetime configured by the **ipv6 neighbor binding stale-lifetime** command.

This example shows how to define an IPv6 snooping policy name as policy1and configures an entry to stay in the binding table for an infinite length of time on a trusted port:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# tracking disable stale-lifetime infinite
Device(config-ipv6-snooping)# end
```

# trusted-port

To configure a port to become a trusted port, use the **trusted-port** command in IPv6 snooping policy mode or ND inspection policy configuration mode. To disable this function, use the **no** form of this command.

**trusted-port**
**no trusted-port**

## Syntax Description

This command has no arguments or keywords.

## Command Default

No ports are trusted.

## Command Modes

ND inspection policy configuration (config-nd-inspection)

IPv6 snooping configuration (config-ipv6-snooping)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

When the **trusted-port** command is enabled, limited or no verification is performed when messages are received on ports that have this policy. However, to protect against address spoofing, messages are analyzed so that the binding information that they carry can be used to maintain the binding table. Bindings discovered from these ports will be considered more trustworthy than bindings received from ports that are not configured to be trusted.

This example shows how to define an NDP policy name as policy1, and configures the port to be trusted:

```
Device> enable
Device# configure terminal
Device(config)# ipv6  nd inspection  policy1
Device(config-nd-inspection)# trusted-port
Device(config-nd-inspection)# end
```

This example shows how to define an IPv6 snooping policy name as policy1, and configures the port to be trusted:

```
Device> enable
Device# configure terminal
Device(config)# ipv6 snooping policy policy1
Device(config-ipv6-snooping)# trusted-port
Device(config-ipv6-snooping)# end
```

# umbrella

To configure the Cisco Umbrella Connector on an interface, use the **umbrella** command in interface configuration mode. To remove this configuration, use the **no** form of this command.

**umbrella** {**in** *tag-name* | **out**}
**no umbrella** {**in** | **out**}

## Syntax Description

| | |
|---|---|
| **in** | Configures the Cisco Umbrella Connector on an interface that is connected to a client. |
| *tag-name* | The interface tag name.<br>The length should not exceed 49 characters. |
| **out** | Configures the Cisco Umbrella Connector on an interface that is used to reach the umbrella server. |

## Command Default

No default behavior or values.

## Command Modes

Interface configuration (config-if)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Amsterdam 17.1.1 | This command was introduced. |

## Usage Guidelines

You should configure the **umbrella out** command before you configure the **umbrella in** command. Registration is successful only when port 443 is in Open state and allows traffic to pass through the existing firewall.

The **umbrella in** and **umbrella out** commands cannot be configured on the same interface.

## Examples

The following example shows how to configure the Cisco Umbrella Connector on an interface:

```
Device> enable
Device# configure terminal
Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# umbrella out
Device(config-if)# exit
Device(config)# interface GigabitEthernet 1/0/2
Device(config-if)# umbrella in mydevice_tag
```

## Related Commands

| Command | Description |
|---|---|
| **show umbrella config** | Displays the Cisco Umbrella Integration configurations of the device. |

# use-updated-eth-header

To enable interoperability between devices and any port on a device that includes the updated Ethernet header in MACsec Key Agreement Protocol Data Units (MKPDUs) for integrity check value (ICV) calculation, use the **ssci-based-on-sci** command in MKA-policy configuration mode. To disable the updated ethernet header in MKPDUs for ICV calculation, use the **no** form of this command.

**use-updated-eth-header**
**no use-updated-eth-header**

## Syntax Description

This command has no arguments or keywords.

## Command Default

The Ethernet header for ICV calculation is disabled.

## Command Modes

MKA-policy configuration (config-mka-policy)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Gibraltar 16.12.1 | This command was introduced. |

## Usage Guidelines

The updated Ethernet header is non-standard. Enabling this option ensures that an MACsec Key Agreement (MKA) session between the devices can be set up.

## Examples

The following example shows how to enable the updated Ethernet header in MKPDUs for ICV calculation:

```
Device> enable
Device# configure terminal
Device(config)# mka policy 2
Device(config-mka-policy)# use-updated-eth-header
```

## Related Commands

| Command | Description |
|---|---|
| **mka policy** | Configures an MKA policy. |
| **confidentiality-offset** | Sets the confidentiality offset for MACsec operations. |
| **delay-protection** | Configures MKA to use delay protection in sending MKPDU. |
| **include-icv-indicator** | Includes ICV indicator in MKPDU. |
| **key-server** | Configures MKA key-server options. |
| **macsec-cipher-suite** | Configures cipher suite for deriving SAK. |
| **sak-rekey** | Configures the SAK rekey interval. |
| **send-secure-announcements** | Configures MKA to send secure announcements in sending MKPDUs. |
| **ssci-based-on-sci** | Computes SSCI based on the SCI. |

# username

To establish the username-based authentication system, use the **username** command in global configuration mode. To remove an established username-based authentication, use the **no** form of this command.

**username** *name* [**aaa attribute list** *aaa-list-name*]
**username** *name* [**access-class** *access-list-number*]
**username** *name* [**algorithm-type** { **md5** { **secret** | **masked-secret** } | **scrypt** { **secret** | **masked-secret** } | **sha256** { **secret** | **masked-secret** }}]
**username** *name* [**autocommand** *command*]
**username** *name* [**callback-dialstring** *telephone-number*]
**username** *name* [**callback-line** [**tty** ]*line-number* [*ending-line-number*]]
**username** *name* [**callback-rotary** *rotary-group-number*]
**username** *name* [**common-criteria-policy** *policy-name*]
**username** *name* [**dnis**]
**username** *name* [**mac**]
**username** *name* [**nocallback-verify**]
**username** *name* [**noescape**]
**username** *name* [**nohangup**]
**username** *name* [**nopassword** | **password** *password* | **password** *encryption-type encrypted-password*]
**username** *name* [**one-time** {**password** {**0** | **6** | **7** | *password*} | **secret** {**0** | **5** | **8** | **9** | *password*}}]
**username** *name* [**password** *secret*]
**username** *name* [**privilege** *level*]
**username** *name* [**secret** {**0** | **5** |*password*}]
**username** *name* [**serial-number**]
**username** *name* [**user-maxlinks** *number*]
**username** *name* [**view** *view-name*]
**no username** *name*

## Syntax Description

| | |
|---|---|
| *name* | Hostname, server name, user ID, or command name. The *name* argument can be only one word. Blank spaces and quotation marks are not allowed. |
| **aaa attribute list** *aaa-list-name* | (Optional) Uses the specified authentication, authorization, and accounting (AAA) method list. |
| **access-class** *access-list-number* | (Optional) Specifies an outgoing access list that overrides the access list specified in the **access-class** command that is available in line configuration mode. It is used for the duration of the user's session. |
| **algorithm-type** | (Optional) Specifies the algorithm to use for hashing the plaintext secret for the user.<br>• **md5**: Encodes the password using the MD5 algorithm.<br>• **scrypt**: Encodes the password using the SCRYPT hashing algorithm.<br>• **sha256**: Encodes the password using the PBKDF2 hashing algorithm.<br>• **secret**: Specifies the secret for the user.<br>• **masked-secret**: Masks the secret input and converts to the selected encryption. |

| | |
|---|---|
| **autocommand** *command* | (Optional) Causes the specified **autocommand**command to be issued automatically after the user logs in. When the specified **autocommand** command is complete, the session is terminated. Because the command can be of any length and can contain embedded spaces, commands using the **autocommand** keyword must be the last option on the line. |
| **callback-dialstring** *telephone-number* | (Optional) Permits you to specify a telephone number to pass to the Data Circuit-terminating Equipment (DCE) device; for asynchronous callback only. |
| **callback-line** *line-number* | (Optional) Specifies relative number of the terminal line (or the first line in a contiguous group) on which you enable a specific username for callback; for asynchronous callback only. Numbering begins with zero. |
| *ending-line-number* | (Optional) Relative number of the last line in a contiguous group on which you want to enable a specific username for callback. If you omit the keyword (such as **tty**), then line number and ending line number are absolute rather than relative line numbers. |
| **tty** | (Optional) Specifies standard asynchronous line; for asynchronous callback only. |
| **callback-rotary** *rotary-group-number* | (Optional) Permits you to specify a rotary group number on which you want to enable a specific username for callback; for asynchronous callback only. The next available line in the rotary group is selected. Range: 1 to 100. |
| **common-criteria-policy** | (Optional) Specifies the name of the common criteria policy. |
| **dnis** | (Optional) Does not require a password when obtained through the Dialed Number Identification Service (DNIS). |
| **mac** | (Optional) Allows a MAC address to be used as the username for MAC filtering done locally. |
| **nocallback-verify** | (Optional) Specifies that authentication is not required for EXEC callback on the specified line. |
| **noescape** | (Optional) Prevents the user from using an escape character on the host to which that user is connected. |
| **nohangup** | (Optional) Prevents Cisco IOS software from disconnecting the user after an automatic command (set up with the **autocommand** keyword) is run. Instead, the user gets another user EXEC prompt. |
| **nopassword** | (Optional) No password is required for the user to log in. This is usually the most useful keyword to use in combination with the **autocommand** keyword. |
| **password** | (Optional) Specifies a password to access the *name* argument. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the **username** command. |
| *password* | Password that the user enters. |

| | |
|---|---|
| *encryption-type* | Single-digit number that defines whether the text immediately following the **password** is encrypted, and if so, what type of encryption is used. Defined encryption types are 0, which means that the text immediately following the **password** is not encrypted, and 6 and 7, which means that the text is encrypted using a Cisco-defined encryption algorithm. |
| *encrypted-password* | Encrypted password that the user enters. |
| **one-time** | (Optional) Specifies that the username and password is valid for only one time. This configuration is used to prevent default credentials from remaining in user configurations.<br>• **0**: Specifies that an unencrypted password or secret (depending on the configuration) follows.<br>• **6**: Specifies that an encrypt password follows.<br>• **7**: Specifies that a hidden password follows.<br>• **5**: Specifies that a MD5 HASHED secret follows.<br>• **8**: Specifies that a PBKDF2 HASHED secret follows.<br>• **9**: Specifies that a SCRYPT HASHED secret follows. |
| **secret** | (Optional) Specifies a secret for the user. |
| *secret* | For Challenge Handshake Authentication Protocol (CHAP) authentication. Specifies the secret for the local device or the remote device. The secret is encrypted when it is stored on the local device. The secret can consist of any string of up to 11 ASCII characters. There is no limit to the number of username and password combinations that can be specified, allowing any number of remote devices to be authenticated. |
| **privilege** *privilege-level* | (Optional) Sets the privilege level for the user. Range: 1 to 15. |
| **serial-number** | (Optional) Specifies the serial number. |
| **user-maxlinks** *number* | (Optional) Specifies the maximum number of inbound links allowed for the user. |
| **view** *view-name* | (Optional) Associates a CLI view name, which is specified with the **parser view** command, with the local AAA database; for CLI view only. |

**Command Default**

No username-based authentication system is established.

**Command Modes**

Global configuration (config)

**Command History**

| Release | Modifica |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This com<br>introduce |
| Cisco IOS XE Dublin 17.10.1 | The **mas**<br>keyword |

## Usage Guidelines

The **username** command provides username or password authentication, or both, for login purposes only.

Multiple **username** commands can be used to specify options for a single user.

Add a username entry for each remote system with which the local device communicates, and from which it requires authentication. The remote device must have a username entry for the local device. This entry must have the same password as the local device's entry for that remote device.

This command can be useful for defining usernames that get special treatment. For example, you can use this command to define an *info* username that does not require a password, but connects the user to a general purpose information service.

The **username** command is required as part of the configuration for CHAP. Add a username entry for each remote system from which the local device requires authentication.

To enable the local device to respond to remote CHAP challenges, one **username** *name* entry must be the same as the **hostname** entry that has already been assigned to the other device. To avoid the situation of a privilege level 1 user entering into a higher privilege level, configure a per-user privilege level other than 1, for example, 0 or 2 through 15. Per-user privilege levels override virtual terminal privilege levels.

### CLI and Lawful Intercept Views

Both CLI views and lawful intercept views restrict access to specified commands and configuration information. A lawful intercept view allows the user to secure access to lawful intercept commands that are held within the TAP-MIB, which is a special set of SNMP commands that store information about calls and users.

Users who are specified via the **lawful-intercept** keyword are placed in the lawful-intercept view by default if no other privilege level or view name is explicitly specified.

If no value is specified for the *secret* argument, and the **debug serial-interface** command is enabled, an error is displayed when a link is established and the CHAP challenge is not implemented. The CHAP debugging information is available using the **debug ppp negotiation**, **debug serial-interface**, and **debug serial-packet** commands.

## Examples

The following example shows how to implement a service similar to the UNIX **who** command, which can be entered at the login prompt, and lists the current users of the device:

```
Device> enable
Device# configure terminal
Device(config)# username who nopassword nohangup autocommand show users
```

The following example shows how to implement an information service that does not require a password to be used:

```
Device> enable
Device# configure terminal
Device(config)# username info nopassword noescape autocommand telnet nic.ddn.mil
```

The following example shows how to implement an ID that works even if all the TACACS+ servers break:

```
Device> enable
Device# configure terminal
Device(config)# username superuser password superpassword
```

The following example shows how to enable CHAP on interface serial 0 of server_l. It also defines a password for a remote server named server_r.

```
hostname server_l
username server_r password theirsystem
interface serial 0
 encapsulation ppp
 ppp authentication chap
```

The following is a sample output from the **show running-config** command displaying the passwords that are encrypted:

```
hostname server_l
username server_r password 7 121F0A18
interface serial 0
 encapsulation ppp
 ppp authentication chap
```

The following example shows how a privilege level 1 user is denied access to privilege levels higher than 1:

```
Device> enable
Device# configure terminal
Device(config)# username user privilege 0 password 0 cisco
Device(config)# username user2 privilege 2 password 0 cisco
```

The following example shows how to remove username-based authentication for user2:

```
Device> enable
Device# configure terminal
Device(config)# no username user2
```

The following example shows how to generate a type 8 (PBKDF2 with SHA-256) masked password:

```
Device> enable
Device# configure terminal
Device(config)# username user1 algorithm-type sha256 masked-secret
Enter secret: ****
Confirm secret: ****
Device(config)#show run | sec username
username user1 secret 8 $8$SmjcLxCNli8lGE$u.vFlaiPqJXBGFaQcEEljsQ/YAxI/LdemFlLoAe3TM
```

## Related Commands

| Command | Description |
|---|---|
| **debug ppp negotiation** | Displays PPP packets sent during PPP startup, where PPP option |
| **debug serial-interface** | Displays information about a serial connection failure. |
| **debug serial-packet** | Displays more detailed serial interface debugging information th using the **debug serial interface** command. |

# vlan access-map

To create or modify a VLAN map entry for VLAN packet filtering, and change the mode to the VLAN access-map configuration, use the **vlan access-map** command in global configuration mode on the device. To delete a VLAN map entry, use the **no** form of this command.

**vlan access-map** *name* [*number*]
**no vlan access-map** *name* [*number*]

## Syntax Description

| | |
|---|---|
| *name* | Name of the VLAN map. |
| *number* | (Optional) The sequence number of the map entry that you want to create or modify (0 to 65535). If you are creating a VLAN map and the sequence number is not specified, it is automatically assigned in increments of 10, starting from 10. This number is the sequence to insert to, or delete from, a VLAN access-map entry. |

## Command Default

There are no VLAN map entries and no VLAN maps applied to a VLAN.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

In global configuration mode, use this command to create or modify a VLAN map. This entry changes the mode to VLAN access-map configuration, where you can use the **match** access-map configuration command to specify the access lists for IP or non-IP traffic to match and use the **action** command to set whether a match causes the packet to be forwarded or dropped.

In VLAN access-map configuration mode, these commands are available:

- **action**—Sets the action to be taken (forward or drop).
- **default**—Sets a command to its defaults.
- **exit**—Exits from VLAN access-map configuration mode.
- **match**—Sets the values to match (IP address or MAC address).
- **no**—Negates a command or set its defaults.

When you do not specify an entry number (sequence number), it is added to the end of the map.

There can be only one VLAN map per VLAN and it is applied as packets are received by a VLAN.

You can use the **no vlan access-map** *name* [*number*] command with a sequence number to delete a single entry.

Use the **vlan filter** interface configuration command to apply a VLAN map to one or more VLANs.

## Examples

This example shows how to create a VLAN map named vac1 and apply matching conditions and actions to it. If no other entries already exist in the map, this will be entry 10.

```
Device> enable
Device# configure terminal
Device(config)# vlan access-map vac1
Device(config-access-map)# match ip address acl1
Device(config-access-map)# action forward
Device(config-access-map)# end
```

This example shows how to delete VLAN map vac1:

```
Device> enable
Device# configure terminal
Device(config)# no vlan access-map vac1
Device(config)# exit
```

# vlan dot1Q tag native

To enable dot1q (IEEE 802.1Q) tagging for a native VLAN on a trunk port, use the **vlan dot1Q tag native** command in global configuration mode.

To disable this function, use the **no** form of this command.

**vlan dot1Q tag native**
**no vlan dot1Q tag native**

## Syntax Description

This command has no arguments or keywords.

## Command Default

Disabled

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

Typically, you configure 802.1Q trunks with a native VLAN ID which strips tagging from all packets on that VLAN.

To maintain the tagging on the native VLAN and drop untagged traffic, use the **vlan dot1q tag native** command. The device will tag the traffic received on the native VLAN and admit only 802.1Q-tagged frames, dropping any untagged traffic, including untagged traffic in the native VLAN.

Control traffic continues to be accepted as untagged on the native VLAN on a trunked port, even when the **vlan dot1q tag native** command is enabled.

**Note** If the **dot1q tag vlan native** command is configured at global level, dot1x reauthentication will fail on trunk ports.

This example shows how to enable dot1q (IEEE 802.1Q) tagging for native VLANs on all trunk ports on a device:

```
Device(config)# vlan dot1q tag native
Device(config)#
```

## Related Commands

| Command | Description |
|---|---|
| **show vlan dot1q tag native** | Displays the status of tagging on the native VLAN. |

# vlan filter

To apply a VLAN map to one or more VLANs, use the **vlan filter** command in global configuration mode. Use the **no** form of this command to remove the map.

**vlan filter** *mapname* **vlan-list** {*list* | **all**}
**no vlan filter** *mapname* **vlan-list** {*list* | **all**}

## Syntax Description

| | |
|---|---|
| *mapname* | Name of the VLAN map entry. |
| **vlan-list** | Specifies which VLANs to apply the map to. |
| *list* | The list of one or more VLANs in the form tt, uu-vv, xx, yy-zz, where spaces around commas and dashes are optional. The range is 1 to 4094. |
| **all** | Adds the map to all VLANs. |

## Command Default

There are no VLAN filters.

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

To avoid accidentally dropping too many packets and disabling connectivity in the middle of the configuration process, we recommend that you completely define the VLAN access map before applying it to a VLAN.

## Examples

This example applies VLAN map entry map1 to VLANs 20 and 30:

```
Device> enable
Device# configure terminal
Device(config)# vlan filter map1 vlan-list 20, 30
Device(config)# exit
```

This example shows how to delete VLAN map entry mac1 from VLAN 20:

```
Device> enable
Device# configure terminal
Device(config)# no vlan filter map1 vlan-list 20
Device(config)# exit
```

You can verify your settings by entering the **show vlan filter** command.

# vlan group

To create or modify a VLAN group, use the **vlan group** command in global configuration mode. To remove a VLAN list from the VLAN group, use the **no** form of this command.

**vlan group** *group-name* **vlan-list** *vlan-list*
**no vlan group** *group-name* **vlan-list** *vlan-list*

## Syntax Description

| | |
|---|---|
| *group-name* | Name of the VLAN group. The group name may contain up to 32 characters and must begin with a letter. |
| **vlan-list** *vlan-list* | Specifies one or more VLANs to be added to the VLAN group. The *vlan-list* argument can be a single VLAN ID, a list of VLAN IDs, or VLAN ID range. Multiple entries are separated by a hyphen (-) or a comma (,). |

## Command Modes

Global configuration (config)

## Command History

| Release | Modification |
|---|---|
| Cisco IOS XE Everest 16.5.1a | This command was introduced. |

## Usage Guidelines

If the named VLAN group does not exist, the **vlan group** command creates the group and maps the specified VLAN list to the group. If the named VLAN group exists, the specified VLAN list is mapped to the group.

The **no** form of the **vlan group** command removes the specified VLAN list from the VLAN group. When you remove the last VLAN from the VLAN group, the VLAN group is deleted.

A maximum of 100 VLAN groups can be configured, and a maximum of 4094 VLANs can be mapped to a VLAN group.

## Examples

This example shows how to map VLANs 7 through 9 and 11 to a VLAN group:

```
Device> enable
Device# configure terminal
Device(config)# vlan group group1 vlan-list 7-9,11
Device(config)# exit
```

This example shows how to remove VLAN 7 from the VLAN group:

```
Device> enable
Device# configure terminal
Device(config)# no vlan group group1 vlan-list 7
Device(config)# exit
```