

# **AOS-CX 10.07 Security Guide**

**8320, 8325, 8360, 8400 Switch Series**



a Hewlett Packard  
Enterprise company

Part Number: 5200-7886a  
Published: October 2021  
Edition: 2

## **Copyright Information**

© Copyright 2021 Hewlett Packard Enterprise Development LP.

## **Open Source Code**

This product includes code licensed under the GNU General Public License, the GNU Lesser General Public License, and/or certain other open source licenses. A complete machine-readable copy of the source code corresponding to such code is available upon request. This offer is valid to anyone in receipt of this information and shall expire three years following the date of the final distribution of this product version by Hewlett Packard Enterprise Company. To obtain such source code, send a check or money order in the amount of US \$10.00 to:

Hewlett Packard Enterprise Company  
6280 America Center Drive  
San Jose, CA 95002  
USA

## **Notices**

The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Confidential computer software. Valid license from Hewlett Packard Enterprise required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Links to third-party websites take you outside the Hewlett Packard Enterprise website. Hewlett Packard Enterprise has no control over and is not responsible for information outside the Hewlett Packard Enterprise website.

<b>Contents</b>	<b>3</b>
<b>About this document</b>	<b>9</b>
Applicable products	9
Latest version available online	9
Command syntax notation conventions	9
About the examples	10
Identifying switch ports and interfaces	10
Identifying modular switch components	11
<b>About security</b>	<b>12</b>
About Authentication, Authorization, and Accounting (AAA)	12
<b>Managing local users and groups</b>	<b>13</b>
Default user admin	13
Example of first login with password setting	13
Built-in user groups and their privileges	13
User-defined user groups	14
User name requirements	14
Password requirements	15
User and user group management tasks	15
Resetting the switch admin password using the Service OS console	16
Prerequisites	16
Procedure	16
Resetting the admin password by reverting the switch to factory defaults	17
Prerequisites	17
Procedure	17
User and group commands	18
user	18
user-group	20
user password	24
service export-password	25
show user-group	26
show user information	27
show user-list	28
<b>SSH server</b>	<b>31</b>
SSH defaults	31
SSH server tasks	31
SSH server commands	32
show ssh host-key	32
show ssh server	33
show ssh server sessions	36
ssh ciphers	37
ssh host-key	39
ssh host-key-algorithms	39
ssh key-exchange-algorithms	41
ssh known-host remove	42
ssh macs	43

ssh maximum-auth-attempts .....	44
ssh public-key-algorithms .....	44
ssh server vrf .....	46
<b>SSH client .....</b>	<b>47</b>
SSH client commands .....	47
ssh (client login) .....	47
<b>Local AAA .....</b>	<b>49</b>
Local AAA defaults and limits .....	49
Local authentication .....	49
Password-based local authentication .....	49
SSH public key-based local authentication .....	50
Local authentication tasks .....	50
Local authorization .....	52
Local authorization tasks .....	52
Local accounting .....	53
Local accounting tasks .....	53
<b>Local AAA commands .....</b>	<b>54</b>
aaa accounting all-mgmt .....	54
aaa authentication console-login-attempts .....	55
aaa authentication limit-login-attempts .....	56
aaa authentication login .....	57
aaa authentication minimum-password-length .....	58
aaa authorization commands .....	59
show aaa accounting .....	61
show aaa authentication .....	61
show aaa authorization .....	62
show ssh authentication-method .....	63
show user .....	64
ssh password-authentication .....	65
ssh public-key-authentication .....	65
user authorized-key .....	66
<b>Remote AAA with TACACS+ .....</b>	<b>68</b>
Default server groups .....	68
Remote AAA (TACACS+) defaults and limits .....	68
About global versus per-TACACS+ server passkeys (shared secrets) .....	69
Remote AAA TACACS+ server configuration requirements .....	69
User role assignment using TACACS+ attributes .....	69
TACACS+ server redundancy and access sequence .....	70
Single source IP address for consistent source identification to AAA servers .....	70
TACACS+ general tasks .....	71
TACACS+ authentication .....	71
About authentication fail-through .....	71
TACACS+ authentication tasks .....	72
TACACS+ authorization .....	72
Using local authorization as fallback from TACACS+ authorization .....	73
About authentication fail-through and authorization .....	73
TACACS+ authorization tasks .....	73
TACACS+ accounting .....	74
Sample accounting information on a TACACS+ server .....	74
Sample REST accounting information on a TACACS+ server .....	75
TACACS+ accounting tasks .....	75
Example: Configuring the switch for Remote AAA with TACACS+ .....	76

Prerequisites .....	76
Procedure .....	76
<b>Remote AAA with RADIUS .....</b>	<b>79</b>
Default server groups .....	79
Remote AAA (RADIUS) defaults and limits .....	79
About global versus per-RADIUS server passkeys (shared secrets) .....	80
Remote AAA RADIUS server configuration requirements .....	80
User role assignment using RADIUS attributes .....	81
RADIUS server redundancy and access sequence .....	81
Single source IP address for consistent source identification to AAA servers .....	82
RADIUS general tasks .....	82
RADIUS authentication .....	83
About authentication fail-through .....	83
RADIUS authentication tasks .....	83
Configuring two-factor authentication .....	84
Prerequisites .....	84
Procedure .....	84
Secure RADIUS (RadSec) .....	85
RadSec configuration .....	86
Deployment scenarios .....	86
Example of RadSec configuration .....	87
RADIUS accounting .....	88
Sample general accounting information .....	89
RADIUS accounting tasks .....	90
Example: Configuring the switch for Remote AAA with RADIUS .....	91
Prerequisites .....	91
Procedure .....	91
<b>Remote AAA (TACACS+, RADIUS) commands .....</b>	<b>94</b>
aaa accounting all-mgmt .....	94
aaa authentication allow-fail-through .....	96
aaa authentication login .....	96
aaa authorization commands .....	98
aaa group server .....	100
radius-server auth-type .....	101
radius-server host .....	102
radius-server host secure ipsec .....	104
radius-server host tls (RadSec) .....	108
radius-server key .....	110
radius-server retries .....	111
radius-server timeout .....	111
radius-server tls timeout (RadSec) .....	112
radius-server tracking .....	113
server .....	114
show aaa accounting .....	116
show aaa authentication .....	118
show aaa authorization .....	120
show aaa server-groups .....	121
show accounting log .....	123
show radius-server .....	126
show radius-server secure ipsec .....	129
show radius-server statistics .....	130
show radius-server statistics host .....	131
show tacacs-server .....	132
show tacacs-server statistics .....	134

show tech aaa .....	135
tacacs-server auth-type .....	138
tacacs-server host .....	139
tacacs-server key .....	141
tacacs-server timeout .....	142
tacacs-server tracking .....	143

## **PKI** ..... **145**

PKI concepts .....	145
Digital certificate .....	145
Certificate authority .....	145
Root certificate .....	145
Leaf certificate .....	146
Intermediate certificate .....	146
Trust anchor .....	146
OCSP .....	146
PKI on the switch .....	146
Trust anchor profiles .....	146
Leaf certificates .....	147
Mandatory matching of peer device hostname .....	147
PKI EST .....	147
EST usage overview .....	147
Prerequisites for using EST for certificate enrollment .....	148
EST profile configuration .....	148
Certificate enrollment .....	148
Certificate re-enrollment .....	148
Checking EST profile and certificate configuration .....	149
EST best practices .....	149
Example using EST for certificate enrollment .....	149
Example including the use of an intermediate certificate .....	155
Installing a self-signed leaf certificate (created inside the switch) .....	157
Procedure .....	157
Installing a self-signed leaf certificate (created outside the switch) .....	158
Prerequisites .....	158
Procedure .....	158
Installing a certificate of a root CA .....	159
Prerequisites .....	159
Procedure .....	159
Installing a CA-signed leaf certificate (initiated in the switch) .....	160
Prerequisites .....	160
Procedure .....	160
Installing a CA-signed leaf certificate (created outside the switch) .....	161
Prerequisites .....	161
Procedure .....	161
PKI commands .....	162
crypto pki application .....	162
crypto pki certificate .....	163
crypto pki ta-profile .....	164
enroll self-signed .....	165
enroll terminal .....	166
import (CA-signed leaf certificate) .....	166
import (self-signed leaf certificate) .....	168
key-type .....	170
ocsp disable-nonce .....	171
ocsp enforcement-level .....	171
ocsp url .....	172

ocsp vrf .....	173
revocation-check ocsp .....	174
show crypto pki application .....	174
show crypto pki certificate .....	175
show crypto pki ta-profile .....	177
ta-certificate .....	178
subject .....	180
PKI EST commands .....	181
arbitrary-label .....	181
arbitrary-label-enrollment .....	182
arbitrary-label-reenrollment .....	183
crypto pki est-profile .....	184
enroll est-profile .....	184
reenrollment-lead-time .....	185
retry-count .....	186
retry-interval .....	187
show crypto pki est-profile .....	187
url .....	188
username .....	189
vrf .....	191
<b>MACsec .....</b>	<b>192</b>
MACsec configuration basics .....	192
MACsec commands .....	193
apply macsec policy .....	193
cipher-suite .....	195
clear macsec statistics .....	196
confidentiality .....	197
include-sci-tag .....	197
macsec policy .....	198
replay-protection .....	199
show macsec policy .....	200
show macsec statistics .....	200
show macsec status .....	203
MKA commands (MACsec) .....	204
apply mka policy .....	204
clear mka statistics .....	206
key-server-priority .....	206
mka policy .....	207
pre-shared-key .....	208
show mka policy .....	209
show mka statistics .....	210
show mka status .....	211
transmit-interval .....	212
<b>Configuring enhanced security .....</b>	<b>213</b>
Configuring enhanced security .....	213
Prerequisites .....	213
Procedure .....	213
password complexity .....	214
Configuring remote logging using SSH reverse tunnel .....	217
Prerequisites .....	217
Procedure .....	218
CLI user session management commands .....	218
cli-session .....	218

---

<b>Fault Monitor</b>	<b>221</b>
Fault monitoring conditions	221
Excessive oversize packets	221
Excessive fragments	221
Excessive CRC errors	221
Excessive TX drops	221
Excessive link flaps	221
Excessive broadcasts	221
Excessive multicasts	222
Excessive collisions	222
Excessive Late Collisions	222
Excessive alignment errors	222
Fault monitor commands	222
(enabling, disabling faults)	222
action	223
apply fault-monitor profile	225
fault-monitor profile	226
show fault-monitor profile	227
show interface fault-monitor profile	228
show interface fault-monitor status	229
show running-config	230
threshold	231
vsx-sync (fault monitor)	233
 <b>Auditors and auditing tasks</b>	 <b>235</b>
Auditing tasks (CLI)	235
Auditing tasks (Web UI)	235
REST requests and accounting logs	236
 <b>Support and Other Resources</b>	 <b>237</b>
Accessing Aruba Support	237
Accessing Updates	237
Aruba Support Portal	237
My Networking	238
Warranty Information	238
Regulatory Information	238
Documentation Feedback	238



This document describes features of the AOS-CX network operating system. It is intended for administrators responsible for installing, configuring, and managing Aruba switches on a network.

## Applicable products

This document applies to the following products:

- Aruba 8320 Switch Series (JL479A, JL579A, JL581A)
- Aruba 8325 Switch Series (JL624A, JL625A, JL626A, JL627A)
- Aruba 8360 Switch Series (JL700A, JL701A, JL702A, JL703A, JL706A, JL707A, JL708A, JL709A, JL710A, JL711A)
- Aruba 8400 Switch Series (JL375A, JL376A)

## Latest version available online

Updates to this document can occur after initial publication. For the latest versions of product documentation, see the links provided in [Support and Other Resources](#).

## Command syntax notation conventions

Convention	Usage
example-text	Identifies commands and their options and operands, code examples, filenames, pathnames, and output displayed in a command window. Items that appear like the example text in the previous column are to be entered exactly as shown and are required unless enclosed in brackets ( [ ] ).
<b>example-text</b>	In code and screen examples, indicates text entered by a user.
Any of the following: <ul style="list-style-type: none"><li>■ <code>&lt;example-text&gt;</code></li><li>■ <i>example-text</i></li><li>■ <code>example-text</code></li><li>■ <i>example-text</i></li></ul>	Identifies a placeholder—such as a parameter or a variable—that you must substitute with an actual value in a command or in code: <ul style="list-style-type: none"><li>■ For output formats where italic text cannot be displayed, variables are enclosed in angle brackets (&lt; &gt;). Substitute the text—including the enclosing angle brackets—with an actual value.</li><li>■ For output formats where italic text can be displayed, variables might or might not be enclosed in angle brackets. Substitute the text including the enclosing angle brackets, if any, with an actual value.</li></ul>
	Vertical bar. A logical OR that separates multiple items from which you can choose only one. Any spaces that are on either side of the vertical bar are included for readability and are not a required part of the command syntax.

Convention	Usage
{ }	Braces. Indicates that at least one of the enclosed items is required.
[ ]	Brackets. Indicates that the enclosed item or items are optional.
... or ...	Ellipsis: <ul style="list-style-type: none"> <li>■ In code and screen examples, a vertical or horizontal ellipsis indicates an omission of information.</li> <li>■ In syntax using brackets and braces, an ellipsis indicates items that can be repeated. When an item followed by ellipses is enclosed in brackets, zero or more items can be specified.</li> </ul>

## About the examples

Examples in this document are representative and might not match your particular switch or environment. The slot and port numbers in this document are for illustration only and might be unavailable on your switch.

### Understanding the CLI prompts

When illustrating the prompts in the command line interface (CLI), this document uses the generic term `switch`, instead of the host name of the switch. For example:

```
switch>
```

The CLI prompt indicates the current command context. For example:

```
switch>
```

Indicates the operator command context.

```
switch#
```

Indicates the manager command context.

```
switch (CONTEXT-NAME)#
```

Indicates the configuration context for a feature. For example:

```
switch(config-if) #
```

Identifies the `interface` context.

### Variable information in CLI prompts

In certain configuration contexts, the prompt may include variable information. For example, when in the VLAN configuration context, a VLAN number appears in the prompt:

```
switch(config-vlan-100) #
```

When referring to this context, this document uses the syntax:

```
switch(config-vlan-<VLAN-ID>) #
```

Where `<VLAN-ID>` is a variable representing the VLAN number.

## Identifying switch ports and interfaces

Physical ports on the switch and their corresponding logical software interfaces are identified using the format:

```
member/slot/port
```

### On the 83xx Switch Series

- *member*: Always 1. VSF is not supported on this switch.
- *slot*: Always 1. This is not a modular switch, so there are no slots.
- *port*: Physical number of a port on the switch.

For example, the logical interface 1/1/4 in software is associated with physical port 4 on the switch.




---

If using breakout cables, the port designation changes to x:y, where x is the physical port and y is the lane when split to 4 x 10G or 4 x 25G. For example, the logical interface 1/1/4:2 in software is associated with lane 2 on physical port 4 in slot 1 on member 1.

---

## On the 8400 Switch Series

- *member*: Always 1. VSF is not supported on this switch.
- *slot*: Specifies physical location of a module in the switch chassis.
  - Management modules are on the front of the switch in slots 1/5 and 1/6.
  - Line modules are on the front of the switch in slots 1/1 through 1/4, and 1/7 through 1/10.
- *port*: Physical number of a port on a line module

For example, the logical interface 1/1/4 in software is associated with physical port 4 in slot 1 on member 1.

## Identifying modular switch components

- Power supplies are on the front of the switch behind the bezel above the management modules. Power supplies are labeled in software in the format: *member/power supply*:
  - *member*: 1.
  - *power supply*: 1 to 4.
- Fans are on the rear of the switch and are labeled in software as: *member/tray/fan*:
  - *member*: 1.
  - *tray*: 1 to 4.
  - *fan*: 1 to 4.
- Fabric modules are not labeled on the switch but are labeled in software in the format: *member/module*:
  - *member*: 1.
  - *member*: 1 or 2.
- The display module on the rear of the switch is not labeled with a member or slot number.

This AOS-CX Switch provides the following security features:

- Local user and group management.
- Authentication, Authorization, and Accounting (AAA), either local (password or SSH public key-based), or remote password-based TACACS+ or RADIUS.
- SSH server. SSH is a cryptographic protocol that encrypts all communication between devices.
- Ability to use enhanced security as described in [Configuring enhanced security](#).
- Making sensitive switch configuration information available for secure export/import between switches. For information, see `service export-password`.

## About Authentication, Authorization, and Accounting (AAA)

- **Authentication:** identifies users, validates their credentials, and grants switch access.
- **Authorization:** controls authenticated users command execution and switch interaction privileges.
- **Accounting:** collects and manages user session activity logs for auditing and reporting purposes.

Local AAA on your Aruba switch provides:

- Authentication using local password or SSH public key.
- Authorization using role-based access control (RBAC), and optionally, using user-defined local user groups with command authorization rules defined per group.
- Accounting of user activity on the switch using accounting logs.

Remote AAA provides the following for your Aruba switch:

- Authentication using remote AAA servers with either TACACS+ or RADIUS.
- Authorization using remote AAA servers with TACACS+ fine-grained command authorization. Local RBAC or local rule-based authorization is also possible.
- Transmission of locally collected accounting information to remote TACACS+ and RADIUS servers.



---

TACACS+ (Terminal Access Controller Access-Control System Plus) and RADIUS (Remote Authentication Dial-In User Service) server software is readily available as either open source or from various vendors.

---



---

For switches that support multiple management modules such as the Aruba 8400, all AAA functionality discussed only applies to the active management module. See also *AAA on switches with multiple management modules* in the *High Availability Guide*.

---

### Default user admin

A factory-default switch comes with a single user named `admin`.

The `admin` user:

- Has an empty password. Press **Enter** in response to the `admin` password prompt. At initial boot, you are prompted to define a password for the `admin` user. Although empty (blank) passwords are allowed, it is recommended that you use strong passwords for all production switches.
- Is a member of the `administrators` group.
- Cannot be removed from the switch.



The switch `admin` user is distinct from the Service OS `admin` user. The Service OS acts as the `bootloader` and recovery operating system. The Service OS has its own CLI.

### Example of first login with password setting

```
switch login: admin
Password:

Please configure the 'admin' user account password.
Enter new password: *****
Confirm new password: *****
switch#
```

### Built-in user groups and their privileges

The switch provides the following built-in user groups with corresponding roles. Each of these roles comes with a set of privileges.

Group/Role	Privileges
<code>administrators</code>	Administrators have full privileges, including: <ul style="list-style-type: none"><li>■ Full CLI access.</li><li>■ Performing firmware upgrades.</li><li>■ Viewing switch configuration information, including sensitive information such as passwords which are displayed as ciphertext.</li><li>■ Performing switch configuration.</li><li>■ Adding/removing user accounts.</li><li>■ Configuring users accounts, including passwords. Once set, a password cannot be deleted or set to empty.</li></ul>

Group/Role	Privileges
	<ul style="list-style-type: none"> <li>REST API: All methods (GET, PUT, POST, DELETE) and switch resources are available. The privilege level for <code>administrators</code> is 15.</li> </ul>
<code>operators</code>	<p>Operators have no switch configuration privileges. Operators are restricted to:</p> <ul style="list-style-type: none"> <li>Basic display-only CLI access.</li> <li>Viewing of nonsensitive switch configuration information.</li> <li>REST API: Other than the <code>\login</code> and <code>\logout</code> resources, only the GET method is available.</li> </ul> <p>The privilege level for <code>operators</code> is 1.</p>
<code>auditors</code>	<p>Auditors are restricted to functions related to auditing only:</p> <ul style="list-style-type: none"> <li>CLI: Access to commands in the auditor context (<code>auditor&gt;</code>) only.</li> <li>Web UI: Access to the <b>System &gt; Log</b> page only.</li> <li>REST API: POST method available for the <code>\login</code> and <code>\logout</code> resources. GET method available for the following resources only: <ul style="list-style-type: none"> <li>Audit log: <code>/logs/audit</code></li> <li>Event log: <code>/logs/event</code></li> </ul> </li> </ul> <p>The privilege level for <code>auditors</code> is 19.</p>

## User-defined user groups

The switch enables you to create up to 29 user-defined local user groups, for the purpose of configuring local authorization. Each of the 29 user-defined groups support up to 1024 CLI command authorization rules that define what CLI commands can be executed by members of the group.

The local user groups with their command execution rules are useful for the following:

- Providing authorization for use with RADIUS servers.
- Providing fallback authorization for use with TACACS+ servers.
- Providing authorization when neither RADIUS or TACACS+ servers are used.

## User name requirements

`<USERNAME>`

Specifies the user name. Requirements:

- Must start with a lowercase letter.
- Can contain numbers and lowercase letters.
- Can include only these three special characters: hyphens ( - ), dots ( . ), and underscores ( \_ ).
- Can have a maximum of 32 characters.
- Cannot be empty.
- Cannot contain uppercase letters.
- Cannot be: `admin`, `root`, or `remote_user`.
- Cannot be Linux reserved names such as:

`daemon`, `bin`, `sys`, `sync`, `proxy`, `www-data`, `backup`, `list`, `irc`, `gnats`, `nobody`, `systemd-bus-proxy`, `sshd`, `messagebus`, `rpc`, `systemd-journal-gateway`, `systemd-journal-remote`, `systemd-journal-upload`,

systemd-timesync, systemd-coredump, systemd-resolve, rpcuser, vagrant, opsd, rdanet, \_lldpd, rdaadmin, rdaweb, docker\_container, tss.

## Password requirements

Passwords must:

- Contain only ASCII characters from hexadecimal 21 to hexadecimal 7E [\x21-\x7E] (decimal 33 to 126). Spaces are not allowed. When the password is entered directly without prompting, the "?" symbol (hexadecimal 3F [\x3F] (decimal 63)) is not permitted.
- Contain at most 32 characters.
- Contain at least the number of characters configured (optionally) for minimum-password-length.



Although empty passwords are supported, it is recommended that you use strong passwords for all production switches.



Only an administrator can change the password of a user assigned to the `operators` role.

## User and user group management tasks

User and user group management common tasks are as follows:

Task	Command or procedure	Example
Creating a user	<code>user</code>	<code>user jamie group administrators password</code>
Changing a user password	<code>user password</code>	<code>user jamie password</code>
Removing a user	<code>user</code>	<code>no user jamie</code>
Setting a user account password	<code>user password</code>	<code>user admin password</code>
Resetting the admin password using the Service OS	<a href="#">(procedure)</a>	
Resetting the admin password by reverting the switch to factory defaults	<a href="#">(procedure)</a>	<code>erase startup-config</code> <code>boot system</code>
Showing a list of all users	<code>show user-list</code>	<code>show user-list</code>
Showing information for the logged-in user	<code>show user information</code>	<code>show user information</code>
Creating a user group	<code>user-group</code>	<code>user-group admuser2</code>

Task	Command or procedure	Example
Adding command authorization rules to a user group	permit or deny (within user-group)	10 deny cli command "show aaa *.*" 20 permit cli command "show *.*"
Adding comments to rules in a user group	comment (within user-group)	10 comment Deny all show aaa commands. 20 comment Permit all other show commands.
Resequencing rules in a user group	resequence (within user-group)	resequence 100 20
Showing a list of all user groups	show user-group	show user-group

## Resetting the switch admin password using the Service OS console

Perform this task only when the switch (Product OS) `admin` user password has been forgotten.

### Prerequisites

- You are connected to the switch through the console port.
- You know the Service OS password (if configured).



If you forget the Service OS password, the only recourse is to zeroize the switch, reverting it to factory defaults. For more information, see *Zeroization* in the *Diagnostics and Supportability Guide*.

### Procedure

1. Reboot the switch.
2. At the boot prompt, select 0. Service OS Console.

```
0. Service OS Console
1. Primary Software Image [XL.01.01.0001]
2. Secondary Software Image [XL.01.01.0002]
```

3. At the Switch Login prompt, enter `admin` and press `Enter`. If prompted for a Service OS password, enter it and press `Enter`.

```
Switch login: admin
Password: *****
Hewlett Packard Enterprise
SVOS>
```

4. At the `SVOS>` prompt, enter `password` and press `Enter`.
5. Enter the new switch (Product OS) password at both password prompts.



```
SVOS> password
Enter password: ****
Confirm password: ****

SVOS>
```

6. Enter `boot` and press `Enter`.

```
SVOS> boot

ServiceOS Information:
  Version: **.10.06.0001
  Build Date: 2020-12-01 14:52:31 PDT
  Build ID: ServiceOS: **.01.01.0001:461519208911:20180301452
  SHA: 46151920891195cdb2267ea6889a3c6cbc3d4193

Boot Profiles:
0. Service OS Console
1. Primary Software Image [**.10.06.0001]
2. Secondary Software Image [**.10.06.0001]

Select profile(primary):
```

7. To boot with the primary switch image press `1` and then `Enter`. To boot with the secondary switch image, press `2` and then `Enter`. If you make no selection for approximately 10 seconds, the switch boots the default image. The default is shown in parentheses to the right of `Select profile`, for example: `Select profile(primary):.`
8. Once in AOS-CX, save the configuration to make the `admin` login user account password setting persistent.

## Resetting the admin password by reverting the switch to factory defaults



This task erases all switch configuration, reverting the switch to its factory default state. Consider using other less-impacting techniques for admin password reset. For example, another administrator user can reset the admin user password to a known value. See also [Resetting the switch admin password using the Service OS console](#).

### Prerequisites

If wanted, you have saved a copy of the switch configuration.

### Procedure

1. At the manager command prompt, enter `erase startup-config`.

```
switch# erase startup-config
```

2. Enter `boot system`, responding `n` to the `Do you want to save the current configuration` prompt and then responding `y` to the `Continue` prompt.

```
switch# boot system
Do you want to save the current configuration (y/n)? n

This will reboot the entire switch and render it unavailable
until the process is complete.
Continue (y/n)? y
The system is going down for reboot.
```

3. At the login prompt, enter `admin` and press `Enter`. The admin password remains empty until it is set.

## User and group commands

### user

#### Syntax

```
user <USERNAME> group {administrators | operators | auditors | <USER-GROUP>}
password [ciphertext <CIPHERTEXT-PASSWORD> | plaintext <PLAINTEXT-PASSWORD>]
```

```
no user <USERNAME>
```

#### Description

Creates a user and adds the user to one of the user groups. Users are given the privileges of their group. For the three built-in user groups (`administrators`, `operators`, `auditors`), the privileges are fixed. For user-defined local user groups, the privileges are defined by the CLI command authorization rules of the group.

When entered without either optional `ciphertext` or `plaintext` parameters, the cleartext password is prompted for twice, with the characters entered masked with "\*" symbols.

The `no` form of this command removes a user account from the switch. The administrator cannot delete the user account from which they are logged in. The `admin` user cannot be deleted.

#### Command context

```
config
```

#### Parameters

<USERNAME>

Specifies the user name. Requirements:

- Must start with a lowercase letter.
- Can contain numbers and lowercase letters.
- Can include only these three special characters: hyphens ( - ), dots ( . ), and underscores ( \_ ).
- Can have a maximum of 32 characters.
- Cannot be empty.
- Cannot contain uppercase letters.
- Cannot be: `admin`, `root`, or `remote_user`.
- Cannot be Linux reserved names such as:

`daemon`, `bin`, `sys`, `sync`, `proxy`, `www-data`, `backup`, `list`, `irc`, `gnats`, `nobody`, `systemd-bus-proxy`, `sshd`, `messagebus`, `rpc`, `systemd-journal-gateway`, `systemd-journal-remote`, `systemd-journal-upload`,

systemd-timesync, systemd-coredump, systemd-resolve, rpcuser, vagrant, opsd, rdanet, \_lldpd, rdaadmin, rdaweb, docker\_container, tss.

group

Selects the local user group to which the new user will be assigned.

administrators | operators | auditors

Selects one of three built-in local user groups.

<USER-GROUP>

Specifies an existing user-defined local user group.

ciphertext <CIPHERTEXT-PASSWORD>

Specifies a ciphertext password. No password prompts are provided and the ciphertext password is validated before the configuration is applied for the user. The variable <CIPHERTEXT-PASSWORD> is Base64 and is typically copied from another switch using the `show running-config` command output and then pasted into this command.



---

The administrator cannot construct ciphertext passwords themselves. The ciphertext is only created by an AOS-CX switch. The ciphertext is created by setting a password for a user with the `user` command. The ciphertext is available for copying from the `show running-config` output and pasting into the configuration on any other AOS-CX switch. The target switch must have the same export password (default or otherwise) as the source switch.

---

plaintext <PLAINTEXT-PASSWORD>

Specifies the password without prompting. The password is visible as cleartext when entered but is encrypted thereafter. Command history does show the password as cleartext.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

- Up to 63 local users can be added, for a total of 64 users including the default user `admin`. A user can belong to only one group.
- The switch ships with the `admin` user account and three built-in local user groups: `administrators`, `operators`, and `auditors`. The `admin` account belongs to the `administrators` group. The Service OS also includes the administrator user `admin`. The two `admin` users are entirely distinct.
- When a local user account is removed, the user loses all active login/SSH sessions. Any calls on the existing REST session with that local user account fail with a permissions issue as soon as the user is deleted. Soon afterwards, the existing REST sessions with the deleted local user account become invalidated. If a user is viewing the GUI while their account is deleted, the user is redirected to the login page within 60 seconds. The home directory associated with the user is also removed from the switch.
- Cleartext passwords (whether entered with prompting or entered directly) must:
  - Contain only ASCII characters from hexadecimal 21 to hexadecimal 7E [`\x21-\x7E`] (decimal 33 to 126). Spaces are not allowed. When the password is entered directly without prompting, the "?" symbol (hexadecimal 3F [`\x3F`] (decimal 63)) is not permitted.
  - Contain at most 32 characters.

- Contain at least the number of characters configured (optionally) for `minimum-password-length`.



---

Although empty passwords are supported, it is recommended that you use strong passwords for all production switches.

---



---

Only an administrator can change the password of a user assigned to the `operators` role.

---

## Examples

Creating local user `jamie` in the `administrators` group with a prompted password:

```
switch(config)# user jamie group administrators password
Adding user jamie
Enter password:*****
Confirm password:*****
```

Creating user `chris` in the existing user-defined local user group `admuser2` with a cleartext password, using direct entry without prompting:

```
switch(config)# user chris group admuser2 password plaintext passWORDxJ|989
```

Creating user `alex` in the `operators` group with a ciphertext password (the ciphertext shown is a placeholder that must be replaced with actual ciphertext):

```
switch(config)# user alex group operators password ciphertext NDcDI2...8igJfA=
```

Removing user `jamie`:

```
switch(config)# no user jamie
User jamie's home directory and active sessions will be deleted.
Do you want to continue [y/n]?y
```

## user-group

### Syntax

```
user-group <GROUP-NAME>
no user-group <GROUP-NAME>
```

### Description

If `<GROUP-NAME>` does not exist, this command creates a local user group and then enters its context. If `<GROUP-NAME>` exists, this command enters the context for the specified `<GROUP-NAME>`. Within the `<GROUP-NAME>` context, several subcommands are available for working with rules that specify what CLI commands are permitted or denied for all members of the local group.

In addition to the three built-in user groups `administrators`, `operators`, and `auditors`, up to 29 user-defined local user groups can be defined. All users can be members of only one of the up to 32 groups.

The no form of this command deletes the specified user group. All members of the deleted group lose all command authorization privilege.



---

Do not causally delete user-defined local user groups without understanding the implications. Although user-defined local user groups can be deleted with the respective members losing all privileges, the three built-in groups `administrators`, `operators`, and `auditors` are always available and their privileges are unchangeable.

---

## Command context

config

## Authority

Administrators or local user group members with execution rights for this command.

## Subcommands

These subcommands are available within the user-defined local user group context (shown in the switch prompt as `config-usr-grp-<GROUP-NAME>`).

```
[<SEQ-NUM>] {permit | deny} cli command "<REGEX>"  
no <SEQ-NUM>
```

Defines a CLI command privilege `permit` or `deny` rule. There is an implicit "`deny .*`" rule at the end of every user-defined group rule list. Members of a user-defined group without any `permit` rules have no CLI command privileges.

The no form of this subcommand deletes the specified (by sequence number) rule from the group.



---

Rule evaluation proceeds from lowest to highest sequence number until the first successful match, resulting in either CLI command permission or denial. Rule evaluation ceases upon first match. Therefore, rules for related CLI commands must be defined in most restrictive to least restrictive order.

---

<SEQ-NUM>

Specifies the CLI command rule sequence number. When omitted, a sequence number that is 10 greater the highest existing sequence number is auto-assigned. When no rules exist, the first auto-assigned sequence number is 10.

{`permit` | `deny`}

Sets the rule type as either `permit` or `deny`. Rule order is important. For example, these two related rules together authorize all `show` commands except for the `show aaa` commands.

```
switch(config-usr-grp-admuser2) #10 deny cli command "show aaa .*"
switch(config-usr-grp-admuser2) #20 permit cli command "show .*"

```

To achieve the wanted effect in this example, the `deny` rule must precede the `permit` rule. These two rules together achieve the following:

- All `show aaa` commands match on rule 10, triggering command denial, and the immediate cessation of further rule evaluation. Matching on rule 20 is never attempted.
- All other `show` commands (excluding `show aaa` commands) match on rule 20 and are therefore permitted.

<REGEX>

Specifies the CLI command matching criteria of the rule. The criteria can be expressed as `.*` which matches all commands. Otherwise, the criteria is expressed as a POSIX-compliant regular expression (regex) string starting with an exact match command token (for example `show`) followed by a regex representing command arguments. The first word must be a string that contains only alphanumeric or hyphen characters.

For example, to allow all commands starting with the word `interface`, the regex must be `"interface .*"` or just `"interface"`. Using `"interface.%"` (without the space) is not supported. For example, `"show .*"` matches every `show` command. Consult the Extended regular expression information available at: [https://pubs.opengroup.org/onlinepubs/9699919799/basedefs/V1\\_chap09.html#tag\\_09\\_04](https://pubs.opengroup.org/onlinepubs/9699919799/basedefs/V1_chap09.html#tag_09_04).

Sample matching criteria	Sample matched CLI command or specifier	Matches
<code>show .*</code>	<code>show accounting log</code>	All <code>show</code> commands
<code>bgp .*</code>	<code>bgp router-id 1.1.1.1</code>	All <code>bgp</code> commands
<code>interface .*</code>	<code>interface 1/1/1</code>	All interface specifiers
<code>vlan (3 4)</code>	<code>vlan 3</code>	VLAN 3 or 4
<code>vlan [1-9]</code>	<code>vlan 5</code>	A single VLAN in the range 1 to 9
<code>vlan ([1-9] 1[0-9])</code>	<code>vlan 19</code>	A single VLAN in the range 1 to 19

```
[<SEQ-NUM>] comment <TEXT-STRING>
no <SEQ-NUM> comment
```

Adds a comment to an existing rule. The `no` form of this subcommand removes an existing comment.

```
switch(config-usr-grp-admuser2)# 10 comment Deny all show aaa commands.
switch(config-usr-grp-admuser2)# 20 comment Permit all other show commands.
switch(config-usr-grp-admuser2)#
switch(config-usr-grp-admuser2)# show running-config current-context
user-group admuser2
  10 comment Deny all show aaa commands.
  10 deny cli command "show aaa .%"
  20 comment Permit all other show commands.
  20 permit cli command "show .%"
```

```
include <GROUP-NAME> [no] include <GROUP-NAME>
```

Include all rules from the specified user-defined `<GROUP-NAME>`. Only one group can be included in the definition of another group. The content of the included group is effectively placed at the top of the rules list in the current group. If the specified `<GROUP-NAME>` does not exist, it is created.

The `no` form of this subcommand removes the specified included group from the current group. The specified included group must exist and must be included in the current group or else an error message is shown.

The name of the included group is shown at the top of the `show user-group` command for the group with the `include`.

In this example, group `admuser1` is included in group `admuser2`. So the `admuser1` rules are evaluated first and then the rules in the `admuser2` group are only evaluated if no CLI command match occurs for the rules in group `admuser1`.

```

switch(config-usr-grp-admuser2)# include admuser1
switch(config-usr-grp-admuser2)# show user-group admuser2
User Group Summary
=====
Name           : admuser2
Type           : configuration
Included Group  : admuser1
Number of Rules : 2
User Group Rules
=====

```

SEQUENCE NUM	ACTION	COMMAND	COMMENT
10	deny	show aaa .*	Deny all show aaa commands.
20	permit	show .*	Permit all other show commands.

**resequence** [**<STARTING-SEQ-NUM>** **<INCREMENT>**]

Resequences the CLI command authorization rules. When entered without the optional parameters the rules are resequenced with a **<STARTING-SEQ-NUM>** of 10 and an **<INCREMENT>** of 10.

**<STARTING-SEQ-NUM>**

Specifies the starting sequence number.

**<INCREMENT>**

Specifies the sequence number increment.

Resequencing the rules to start at 100 with an increment of 20:

```

switch(config-usr-grp-admuser2)# resequence 100 20
switch(config-usr-grp-admuser2)# show running-config current-context
user-group admuser2
  100 comment Deny all show aaa commands.
  100 deny cli command "show aaa .*"
  120 comment Permit all other show commands.
  120 permit cli command "show .*"

```

Resequencing the rules to the default of starting at 10 with an increment of 10:

```

switch(config-usr-grp-admuser2)# resequence
switch(config-usr-grp-admuser2)# show running-config current-context
user-group admuser2
  10 comment Deny all show aaa commands.
  10 deny cli command "show aaa .*"
  20 comment Permit all other show commands.
  20 permit cli command "show .*"

```

**show running-config current-context**

Shows all the commands used to configure the rules in the current group context.

```

switch(config-usr-grp-admuser2)# show running-config current-context
user-group admuser2
  10 comment Deny all show aaa commands.
  10 deny cli command "show aaa .*"
  20 comment Permit all other show commands.
  20 permit cli command "show .*"

```

**list**

List the subcommands available within the user-defined group context.

exit

Exits the user-defined group context.

end

Exits the user-defined group context and then the config context.

## user password

### Syntax

```
user <USERNAME> password [ciphertext <CIPHERTEXT-PASSWORD> | plaintext <PLAINTEXT-PASSWORD>]
```

### Description

Changes a password for an account or enables the password for the admin account. When entered without either optional `ciphertext` or `plaintext` parameters, the cleartext password is prompted for twice, with the characters entered masked with "\*" symbols.

### Command context

config

### Parameters

<USERNAME>

Specifies the corresponding user name for the password you want to change.

ciphertext <CIPHERTEXT-PASSWORD>

Specifies a ciphertext password. No password prompts are provided and the ciphertext password is validated before the configuration is applied for the user. The variable <CIPHERTEXT-PASSWORD> is Base64 and is typically copied from another switch using the `show running-config` command output and then pasted into this command.



---

The administrator cannot construct ciphertext passwords themselves. The ciphertext is only created by an AOS-CX switch. The ciphertext is created by setting a password for a user with the `user` command. The ciphertext is available for copying from the `show running-config` output and pasting into the configuration on any other AOS-CX switch. The target switch must have the same export password (default or otherwise) as the source switch.

---

plaintext <PLAINTEXT-PASSWORD>

Specifies the password without prompting. The password is visible as cleartext when entered but is encrypted thereafter. Command history does show the password as cleartext.

### Authority

Administrators or local user group members with execution rights for this command.

### Usage

The admin account is available on the switch without a password by default.

Cleartext passwords (whether entered with prompting or entered directly) must:

- Contain only ASCII characters from hexadecimal 21 to hexadecimal 7E [\x21-\x7E] (decimal 33 to 126). Spaces are not allowed. When the password is entered directly without prompting, the "?" symbol (hexadecimal 3F [\x3F] (decimal 63)) is not permitted.
- Contain at most 32 characters.



- Contain at least the number of characters configured (optionally) for `minimum-password-length`.



---

Although empty passwords are supported, it is recommended that you use strong passwords for all production switches.

---



---

Only an administrator can change the password of a user assigned to the `operators` role.

---

## Examples

Enabling (or changing) a cleartext password for `admin`:

```
switch(config)# user admin password
Changing password for user admin
Enter password:*****
Confirm password:*****
```

Changing the cleartext password for user `chris`, using direct entry without prompting:

```
switch(config)# user chris password plaintext PASSwordZQ#@67
```

Changing the ciphertext password for user `alex` (the ciphertext shown is a placeholder that must be replaced with actual ciphertext):

```
switch(config)# user alex password ciphertext XqYJ36...W83D4Y=
```

## service export-password

### Syntax

```
service export-password
no service export-password
```

### Description

Configures a nondefault export password. The export password is used to transform critical security parameters (such as password hashes) into ciphertext suitable for exporting and showing by commands such as `show running-config`. This transformation enables safe switch configuration import and export. The `no` form of this command reverts the export password to its factory default.



---

All factory-default switches have identical default export passwords. For security, it is recommended that you set the same nondefault export password on every switch in a group that will exchange configuration information. Only switches with identical export passwords can exchange configuration information.

---

### Command context

`config`

### Authority

Administrators or local user group members with execution rights for this command.

## Usage

Prompts you twice for the new export password.

The export password must:

- Contain only ASCII characters from hexadecimal 21 to hexadecimal 7E [\x21-\x7E] (decimal 33 to 126). Spaces are not allowed.
- Contain at most 32 characters.
- Not be blank.

## Examples

Configuring a new export password:

```
switch(config)# service export-password
Enter password:*****
Confirm password:*****
```

Reverting the export password to its factory default:

```
switch(config)# no service export-password
```

## show user-group

### Syntax

```
show user-group [<GROUP-NAME>] [vsx-peer]
```

### Description

Shows user group information for the built-in groups plus any user-defined local user groups. When entered without <GROUP-NAME>, summary information is shown for all groups.

### Command context

Manager (#)

### Parameters

<GROUP-NAME>

Narrows the `show` command output to that of the specified group, and for local user groups, adds the User Group Rules list.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Show the list of all user groups, including built-in groups and local user groups.

```
switch# show user-group
GROUP NAME      GROUP TYPE      INCLUDED GROUP      NUMBER OF RULES
-----
administrators  built-in        n/a                 n/a
admuser1        configuration    --                 5
admuser2        configuration    admuser1            2
auditors        built-in        n/a                 n/a
operators       built-in        n/a                 n/a
```

Show detailed information for local user group `admuser2`.

```
switch(config-usr-grp-admuser2)# show user-group admuser2
User Group Summary
=====
Name           : admuser2
Type           : configuration
Included Group  : admuser1
Number of Rules : 2
User Group Rules
=====
SEQUENCE NUM  ACTION      COMMAND          COMMENT
-----
---
10            deny        show aaa .*      Deny all show aaa commands.
20            permit      show .*          Permit all other show
commands.
```

## show user information

### Syntax

show user information

### Description

Shows the following information for the logged-in user:

- User name.
- User authentication type: `local`, `RADIUS`, or `TACACS+`.
- User group: `administrators`, `operators`, or `<GROUP-NAME>`.
- User privilege level: For the built-in user groups and `RADIUS` or `TACACS+`, the role privilege level value is shown. For user-defined user groups, `N/A` is shown.

### Command context

Operator (>) or Manager (#)

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Examples

Showing information for the `admin` user:

```
switch# show user information
Username           : admin
Authentication type : Local
User group         : administrators
User privilege level : 15
```

Showing information for a member of the user-defined local user group `admuser2`:

```
switch# show user information
Username           : admin2-b
Authentication type : Local
User group         : admuser2
User privilege level : N/A
```

Showing information for a member of `operators`:

```
switch# show user information
Username           : operator
Authentication type : Local
User group         : operators
User privilege level : 1
```

Showing information for remote RADIUS user `rad_user1` mapped to local user group `administrators`:

```
switch# show user information
Username           : rad_user1
Authentication type : RADIUS
User group         : administrators
User privilege level : 15
```

Showing information for remote RADIUS user `rad_user2` mapped to local user group `operators`:

```
switch# show user information
Username           : rad_user2
Authentication type : RADIUS
User group         : operators
User privilege level : 1
```

Showing information for remote TACACS+ `tac_user1` logged in with `priv-lvl 15` (mapped to user group `administrators`):

```
switch# show user information
Username           : tac_user1
Authentication type : TACACS+
User group         : administrators
User privilege level : 15
```

## show user-list

### Syntax

```
show user-list [vsx-peer]
```

### Description

Shows all configured users and their corresponding group names.

## Command context

Manager (#)

## Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Show the user list from a switch with only the admin user defined.

```
switch# show user-list
```

USER	GROUP
admin	administrators

Show the user list after adding a user to the operators built-in group.

```
switch# show user-list
```

USER	GROUP
admin	administrators
oper1	operators

Show the user list after adding a user to the auditors built-in group.

```
switch# show user-list
```

USER	GROUP
admin	administrators
oper1	operators
audit1	auditors

Show the user list after adding a total of three users to two user-defined user groups.

```
switch# show user-list
```

USER	GROUP
admin	administrators
oper1	operators
audit1	auditors
adm1a	admuser1

admin2-a  
admin2-b

admuser2  
admuser2

SSH (Secure Shell) is a cryptographic protocol that encrypts all communication between devices.

Each switch VRF includes an SSH server. The SSH server on the `mgmt` VRF is enabled by default in software version 10.02 and higher, and disabled in version 10.01 and lower. Only the SSH servers included in the switch are supported.

The SSH server provides SSH client to switch communications, enabling SSH clients (at least SSH v2.0) to connect to the switch for the purpose of managing it. The SSH server interfaces with the authentication service that provides local and/or remote AAA.



The SSH server will perform a rekey operation for all open SSH sessions at every hour or after 1 GB of data transferred, whichever occurs first. The rekey is performed to address a common security concern that encryption/decryption keys not be used for long periods of time. This limits the amount of data exposed in the unfortunate case where a key is exposed or refactored.



SSH public key authentication is separate from SSH server. Look for information on *SSH public key* under [Local authentication](#).

## SSH defaults

Setting	Default value
Maximum SSH password retries	3 password retries.
Password-based (with SSH client) authentication	Enabled.
SSH password-based login grace period timeout	120 seconds.
SSH public key authentication	Enabled.
SSH idle session timeout	60 seconds.

## SSH server tasks

SSH server tasks are as follows:

Task	Command name	Example
Enabling the SSH server	<code>ssh server vrf</code>	<code>ssh server vrf default</code>

Task	Command name	Example
Disabling the SSH server	<code>no ssh server vrf</code>	<code>no ssh server vrf default</code>
Generating an SSH host-key pair	<code>ssh host-key</code>	<code>ssh host-key rsa bits 2048</code>
Clearing the list of trusted SSH servers for your user account	<code>ssh known-host remove</code>	<code>ssh known-host remove 1.1.1.1</code>
Configuring SSH to use a set of ciphers	<code>ssh ciphers</code>	<code>ssh ciphers chacha20-poly1305@openssh.com aes256-ctr aes256-cbc</code>
Configuring SSH to use a set of host key algorithms	<code>ssh host-key-algorithms</code>	<code>ssh host-key-algorithms ssh-rsa ssh-ed25519 ecdsa-sha2-nistp521</code>
Configuring SSH to use a set of MACs	<code>ssh macs</code>	<code>ssh macs hmac-sha2-256 hmac-sha2-512</code>
Configuring SSH to use a set of key exchange algorithms	<code>ssh key-exchange-algorithms</code>	<code>ssh key-exchange-algorithms ecdh-sha2- nistp256</code>
Configuring SSH to use a set of public key algorithms	<code>ssh public-key-algorithms</code>	<code>ssh public-key-algorithms x509v3-ssh-rsa ssh-rsa rsa-sha2-256</code>
Configuring SSH idle session timeout	<code>cli-session</code>	<code>switch(config)# cli-session switch(config-cli-session)# timeout 20</code>
Showing the SSH server configuration	<code>show ssh server</code>	<code>show ssh server all-vrfs</code>
Showing the active SSH sessions	<code>show ssh server sessions</code>	<code>show ssh server sessions all-vrfs</code>
Showing the SSH server host keys	<code>show ssh host-key</code>	<code>show ssh host-key ecdsa</code>

## SSH server commands

### show ssh host-key

#### Syntax

```
show ssh host-key [ecdsa | ed25519 | rsa]
```

#### Description

Shows the public host keys for the SSH server. If the key type is not provided, all available host-keys are shown.

#### Command context

Manager (#)

#### Parameters



ecdsa

Selects the ECDSA host-key pair.

ed25519

Selects the ED25519 host-key pair.

rsa

Selects the RSA host-key pair.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Showing the ECDSA public host-key:

```
switch# show ssh host-key ecdsa

Key Type : ECDSA      Curve : ecdsa-sha2-nistp256

ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAhtuv5rABBBGs
...
O4mjVFGMVkZ87RWkyrxeQa2fAGZZEp1902K33/k3q17fA4EivRzC75YvjDu8=
```

Showing all public host keys:

```
switch# show ssh host-key

Key Type : ECDSA      Curve : ecdsa-sha2-nistp256
ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAhtuv5rABBBGs
...
O4mjVFGMVkZ87RWkyrxeQa2fAGZZEp1902K33/k3q17fA4EivRzC75YvjDu8=

Key Type : ED25519
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIGb6910Jwoe8Hkl9K5YhqijrWI3yovNbiJVq6tw4WjJr4

Key Type : RSA        Key Size : 2048
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDdVCXlw43h4n1bwg9jI6DSBMngymCdPD0JUG42Sn9IS
...
nGSXtrNy6OmlFDJTAy+zz5Kd8d21ZLuhf07IHNgF3pff65Xc8qNJBv
```

## show ssh server

### Syntax

```
show ssh server [vrf <VRF-NAME> | all-vrfs] [vsx-peer]
```

### Description

Shows the SSH server configuration for the specified VRF. Administrators can show the server configuration of all VRFs by using the `all-vrfs` parameter. If no VRF name is provided in this command, the command shows the SSH server configuration on the default VRF.

### Command context

Operator (>) or Manager (#)

### Parameters

Selects all VRFs.

Specifies the VRF name.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Operators or Administrators or local user group members with execution rights for this command.  
Operators can execute this command from the operator context (>) only.

## Examples

Showing the SSH server configuration on the default VRF:

```
switch# show ssh server

SSH server configuration on VRF default :

IP Version      : IPv4 and IPv6      SSH Version      : 2.0
TCP Port        : 22                  Grace Timeout (sec) : 120
Max Auth Attempts : 6

Ciphers:
chacha20-poly1305@openssh.com, aes128-ctr, aes192-cbc,
aes128-cbc, aes192-ctr, aes256-gcm@openssh.com,
aes128-gcm@openssh.com, aes256-ctr, aes256-cbc

Host Key Algorithms:
ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521,
ssh-ed25519, rsa-sha2-256, rsa-sha2-512, ssh-rsa

Key Exchange Algorithms:
curve25519-sha256, curve25519-sha256@libssh.org,
ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521,
diffie-hellman-group-exchange-sha256, diffie-hellman-group16-sha512,
diffie-hellman-group18-sha512, diffie-hellman-group14-sha256,
diffie-hellman-group14-sha1

MACs:
hmac-sha1-etm@openssh.com, umac-64@openssh.com,
umac-128@openssh.com, hmac-sha2-256, hmac-sha2-512, hmac-sha1

Public Key Algorithms:
rsa-sha2-256, rsa-sha2-512, ssh-rsa, ecdsa-sha2-nistp256,
ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, ssh-ed25519,
x509v3-rsa2048-sha256, x509v3-ssh-rsa, x509v3-sign-rsa,
x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384,
x509v3-ecdsa-sha2-nistp521
```

Showing the SSH server configuration on the management VRF:

```
switch# show ssh server vrf mgmt

SSH server configuration on VRF mgmt :

IP Version      : IPv4 and IPv6      SSH Version      : 2.0
TCP Port        : 22                  Grace Timeout (sec) : 120
```

```

Max Auth Attempts      : 6

Ciphers:
chacha20-poly1305@openssh.com, aes128-ctr, aes192-cbc,
aes128-cbc, aes192-ctr, aes256-gcm@openssh.com,
aes128-gcm@openssh.com, aes256-ctr, aes256-cbc

Host Key Algorithms:
ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521,
ssh-ed25519, rsa-sha2-256, rsa-sha2-512, ssh-rsa

Key Exchange Algorithms:
curve25519-sha256, curve25519-sha256@libssh.org,
ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521,
diffie-hellman-group-exchange-sha256, diffie-hellman-group16-sha512,
diffie-hellman-group18-sha512, diffie-hellman-group14-sha256,
diffie-hellman-group14-sha1

MACs:
hmac-sha1-etm@openssh.com, umac-64@openssh.com,
umac-128@openssh.com, hmac-sha2-256, hmac-sha2-512, hmac-sha1

Public Key Algorithms:
rsa-sha2-256, rsa-sha2-512ssh-rsa, ecdsa-sha2-nistp256,
ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, ssh-ed25519,
x509v3-rsa2048-sha256, x509v3-ssh-rsa, x509v3-sign-rsa,
x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384,

```

Showing the SSH server configuration for all VRFs:

```

switch# show ssh server all-vrfs

SSH server configuration on VRF default :

IP Version      : IPv4 and IPv6      SSH Version      : 2.0
TCP Port        : 22                  Grace Timeout (sec) : 120
Max Auth Attempts : 6

Ciphers:
chacha20-poly1305@openssh.com, aes128-ctr, aes192-cbc,
aes128-cbc, aes192-ctr, aes256-gcm@openssh.com,
aes128-gcm@openssh.com, aes256-ctr, aes256-cbc

Host Key Algorithms:
ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521,
ssh-ed25519, rsa-sha2-256, rsa-sha2-512, ssh-rsa

Key Exchange Algorithms:
curve25519-sha256, curve25519-sha256@libssh.org,
ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521,
diffie-hellman-group-exchange-sha256, diffie-hellman-group16-sha512,
diffie-hellman-group18-sha512, diffie-hellman-group14-sha256,

MACs:
hmac-sha1-etm@openssh.com, umac-64@openssh.com,
umac-128@openssh.com, hmac-sha2-256, hmac-sha2-512, hmac-sha1

Public Key Algorithms:
rsa-sha2-256, rsa-sha2-512ssh-rsa, ecdsa-sha2-nistp256,

```

```
ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, ssh-ed25519,  
x509v3-rsa2048-sha256, x509v3-ssh-rsa, x509v3-sign-rsa,  
x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384,  
x509v3-ecdsa-sha2-nistp521
```

SSH server configuration on VRF mgmt :

```
IP Version           : IPv4 and IPv6      SSH Version          : 2.0  
TCP Port             : 22                 Grace Timeout (sec)  : 120  
Max Auth Attempts    : 6
```

Ciphers:

```
chacha20-poly1305@openssh.com, aes128-ctr, aes192-cbc,  
aes128-cbc, aes192-ctr, aes256-gcm@openssh.com,  
aes128-gcm@openssh.com, aes256-ctr, aes256-cbc
```

Host Key Algorithms:

```
ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521,  
ssh-ed25519, rsa-sha2-256, rsa-sha2-512, ssh-rsa
```

Key Exchange Algorithms:

```
curve25519-sha256, curve25519-sha256@libssh.org,  
ecdh-sha2-nistp256, ecdh-sha2-nistp384, ecdh-sha2-nistp521,  
diffie-hellman-group-exchange-sha256, diffie-hellman-group16-sha512,  
diffie-hellman-group18-sha512, diffie-hellman-group14-sha256,  
diffie-hellman-group14-sha1
```

MACs:

```
hmac-sha1-etm@openssh.com, umac-64@openssh.com,  
umac-128@openssh.com, hmac-sha2-256, hmac-sha2-512, hmac-sha1
```

Public Key Algorithms:

```
rsa-sha2-256, rsa-sha2-512ssh-rsa, ecdsa-sha2-nistp256,  
ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, ssh-ed25519,  
x509v3-rsa2048-sha256, x509v3-ssh-rsa, x509v3-sign-rsa,  
x509v3-ecdsa-sha2-nistp256, x509v3-ecdsa-sha2-nistp384,
```

## show ssh server sessions

### Syntax

```
show ssh server sessions [vrf <VRF-NAME> | all-vrfs] [vsx-peer]
```

### Description

Shows the active SSH sessions on a specified VRF or on all VRFs. If no VRF is specified, the active sessions on the default VRF are shown.

### Command context

Operator (>) or Manager (#)

### Parameters

all-vrfs

Selects all VRFs.

vrf <VRF-NAME>

Specifies the VRF name.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

If you provide the command with a VRF name, the command shows the active SSH session for the specified VRF. Any user can show sessions of all VRFs by using the `all-vrfs` parameter. The maximum number of sessions per VRF is five. The maximum SSH idle session timeout is 60 seconds.

## Examples

Showing the active SSH sessions on the default VRF:

```
switch# show ssh server sessions

SSH sessions on VRF default
IPv4 SSH Sessions
  Server IP       : 10.1.1.1
  Client IP       : 10.1.1.2
  Client Port     : 58835

IPv6 SSH Sessions
  Server IP       : FF01:0:0:0:0:0:0:FB
  Client IP       : FF01:0:0:0:0:0:0:FC
  Client Port     : 58836
```

Showing the SSH server configuration for all VRFs:

```
switch# show ssh server sessions all-vrf

SSH sessions on VRF mgmt
IPv4 SSH Sessions
  Server IP       : 10.1.1.1
  Client IP       : 10.1.1.2
  Client Port     : 58835

IPv6 SSH Sessions
  Server IP       : FF01:0:0:0:0:0:0:FB
  Client IP       : FF01:0:0:0:0:0:0:FC
  Client Port     : 58836

SSH sessions on VRF default
IPv4 SSH Sessions
  Server IP       : 20.1.1.1
  Client IP       : 20.1.1.2
  Client Port     : 58837

IPv6 SSH Sessions
  Server IP       : FF01:0:0:0:0:0:0:FD
  Client IP       : FF01:0:0:0:0:0:0:FE
  Client Port     : 58838
```

## ssh ciphers

### Syntax

```
ssh ciphers <CIPHERS-LIST>
no ssh ciphers
```

## Description

Configures SSH to use a set of ciphers in the specified priority order. Ciphers in SSH are used for privacy of data being transported over the connection. The first cipher type entered in the CLI is considered a first priority. Each option is an algorithm that is used to encrypt the link and each name indicates the algorithm and cryptographic parameters that are used. Only ciphers that are entered by the user are configured.

The `no` form of this command removes the configuration of ciphers and reverts SSH to use the default set of ciphers.

## Command context

config

## Parameters

<CIPHERS-LIST>

Valid cipher types are:

- aes128-cbc
- aes192-cbc
- aes256-cbc
- aes128-ctr
- aes192-ctr
- aes256-ctr
- aes128-gcm@openssh.com
- aes256-gcm@openssh.com
- chacha20-poly1305@openssh.com

Default set of ciphers in priority order:

1. chacha20-1305@openssh.com
2. aes128-ctr
3. aes192-ctr
4. aes256-ctr
5. aes128-gcm@openssh.com
6. aes256-gcUm@openssh.com

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring SSH to use only specified ciphers in the priority order:

```
switch(config)# ssh ciphers chacha20-poly1305@openssh.com aes256-ctr aes256-cbc
```

Reverting SSH to use the default set of ciphers:

```
switch(config)# no ssh ciphers
```

# ssh host-key

## Syntax

```
ssh host-key {ecdsa [ecdsa-sha2-nistp256 | ecdsa-sha2-nistp384 | ecdsa-sha2-nistp521] |  
ed25519 | rsa [bits {2048 | 4096}] }
```

## Description

Generates an SSH host-key pair.

## Command context

config

## Parameters

ecdsa

Selects the ECDSA host-key pair type as `ecdsa-sha2-nistp256` (the default), `ecdsa-sha2-nistp384`, or `ecdsa-sha2-nistp521`.

ed25519

Selects the ED25519 host-key pair.

rsa

Selects the RSA host-key pair. Optionally, the key bit length is selected with either `bits 2048` (the default) or `bits 4096`.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

When an SSH server is enabled on a VRF for the first time, host-keys are generated.

If the host-key of the given type exists, a warning message is displayed with a request to overwrite the previous host-key with the new key.

## Examples

Overwriting an old ECDSA host-key with a new `ecdsa-sha2-nistp384` host-key:

```
switch(config)# ssh host-key ecdsa ecdsa-sha2-nistp384  
ecdsa host-key will be overwritten.  
Do you want to continue (y/n)?
```

Overwriting an old RSA host-key with a new RSA host-key with 2048 bits:

```
switch(config)# ssh host-key rsa bits 2048  
rsa host-key will be overwritten.  
Do you want to continue (y/n)?
```

Overwriting an ECDSA host-key with an ED25519 host-key pair:

```
switch(config)# ssh host-key ed25519  
ed25519 host-key will be overwritten.  
Do you want to continue (y/n)?
```

## ssh host-key-algorithms

## Syntax

```
ssh host-key-algorithms <HOST-KEY-ALGORITHMS-LIST>
no ssh host-key-algorithms
```

## Description

Configures SSH to use a set of host key algorithms in the specified priority order. Host key algorithms specify which host key types are allowed to be used for the SSH connection. The first host key entered in the CLI is considered a first priority. Each option represents a type of key that can be used. Host keys are used to verify the host that you are connecting to. This configuration allows you to control which host key types are presented to incoming clients, or which host key types to receive first from hosts. Only the host key algorithms that are specified by the user are configured.

The `no` form of this command removes the configuration of host key algorithms and reverts SSH to use the default set of algorithms.

## Command context

config

## Parameters

<HOST-KEY-ALGORITHMS-LIST>

Valid host key algorithms are:

- `ecdsa-sha2-nistp256`
- `ecdsa-sha2-nistp384`
- `ecdsa-sha2-nistp521`
- `rsa-sha2-256`
- `rsa-sha2-512`
- `ssh-ed25519`
- `ssh-rsa`

Default set of host key algorithms in priority order:

1. `ecdsa-sha2-nistp256`
2. `ecdsa-sha2-nistp384`
3. `ecdsa-sha2-nistp521`
4. `ssh-ed25519`
5. `rsa-sha2-256`
6. `rsa-sha2-512`
7. `ssh-rsa`

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring SSH to use only specified host key algorithms:

```
switch(config)# ssh host-key-algorithms ssh-rsa ssh-ed25519 ecdsa-sha2-nistp521
```

Reverting SSH to use the default set of host key algorithms:



```
switch(config)# no host-key-algorithms
```

## ssh key-exchange-algorithms

### Syntax

```
ssh key-exchange-algorithms <KEY-EXCHANGE-ALGORITHMS-LIST>  
no ssh key-exchange-algorithms
```

### Description

Configures SSH to use a set of key exchange algorithm types in the specified priority order. The first key exchange type entered in the CLI is considered a first priority. Key exchange algorithms are used to exchange a shared session key with a peer securely. Each option represents an algorithm that is used to distribute a shared key in a way that prevents outside interference, manipulation, or recovery. Only the key exchange algorithms that are specified by the user are configured.

The `no` form of this command removes the configuration of key exchange algorithms and reverts SSH to use the default set of algorithms.

### Command context

config

### Parameters

<KEY-EXCHANGE-ALGORITHMS-LIST>

Valid key exchange algorithms are:

- curve25519-sha256
- curve25519-sha256@libssh.org
- diffie-hellman-group-exchange-sha1
- diffie-hellman-group-exchange-sha256
- diffie-hellman-group14-sha1
- diffie-hellman-group14-sha256
- diffie-hellman-group16-sha512
- diffie-hellman-group18-sha512
- ecdh-sha2-nistp256
- ecdh-sha2-nistp384
- ecdh-sha2-nistp521

Default set of key exchange algorithms in priority order:

1. curve25519-sha256
2. curve25519-sha256@libssh.org
3. ecdh-sha2-nistp256
4. ecdh-sha2-nistp384
5. ecdh-sha2-nistp521
6. diffie-hellman-group-exchange-sha256
7. diffie-hellman-group16-sha512
8. diffie-hellman-group18-sha512
9. diffie-hellman-group14-sha256

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring SSH to use a set of specified key exchange algorithms:

```
switch(config)# ssh key-exchange-algorithms ecdh-sha2-nistp256 curve25519-sha256
diffie-hellman-group-exchange-sha256
```

Reverting SSH to use the default set of key-exchange-algorithms:

```
switch(config)# no key-exchange-algorithms
```

## ssh known-host remove

### Syntax

```
ssh known-host remove {all | {<IPv4-ADDRESS> | <HOSTNAME> | <IPv6-ADDRESS>} }
```

### Description

Clears the list of trusted SSH servers for your user account. When you download or upload a file to or from a server using SFTP, you establish a trusted SSH relationship with that server. Each user account maintains its own set of SSH server host-keys for every server to which the user previously connected.

### Command context

config

### Parameters

all

Clears the trusted servers list.

<IPv4-ADDRESS>

Specifies the IPv4 address of the remote device.

<HOSTNAME>

Specifies the host name of the remote device. The length of the host name can be up to 255 characters.

<IPv6-ADDRESS>

Specifies the IPv6 address of the remote device.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Clearing the trusted server list:

```
switch(config)# ssh known-host remove all
```

Removing a specified server from the trusted server list:

```
switch(config)# ssh known-host remove 1.1.1.1
```

## ssh macs

### Syntax

```
ssh macs <MACS-LIST>  
no ssh macs
```

### Description

Configures SSH to use a set of message authentication codes (MACs) in the specified priority order. The first MAC entered in the CLI is considered a first priority. MACs maintain the integrity of each message sent across an SSH connection. Each option represents an algorithm that can be used to provide integrity between peers. Only the MAC types that are specified by the user are configured.

The `no` form of this command removes the configuration of MACs and reverts SSH to use the default set of MACs.

### Command context

config

### Parameters

<MACS-LIST>

Valid MAC types are:

- hmac-sha1
- hmac-sha1-96
- hmac-sha1-etm@openssh.com
- hmac-sha2-256
- hmac-sha2-512
- hmac-sha2-256-etm@openssh.com
- hmac-sha2-512-etm@openssh.com

Default set of MACs in priority order:

1. hmac-sha2-256-etm@openssh.com
2. hmac-sha2-512-etm@openssh.com
3. hmac-sha1-etm@openssh.com
4. hmac-sha2-256
5. hmac-sha2-512
6. hmac-sha1

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Configuring SSH to use a set of specified MACs:

```
switch(config)# ssh macs hmac-sha2-256 hmac-sha2-512
```

Reverting SSH to use the default set of MACs:

```
switch(config)# no ssh macs
```

## ssh maximum-auth-attempts

### Syntax

```
ssh maximum-auth-attempts <ATTEMPTS>  
no maximum-auth-attempts
```

### Description

Sets the SSH maximum number of authentication attempts.

The `no` form of the command resets the maximum to its default of 6.

### Command context

config

### Parameters

<ATTEMPTS>

Specifies the maximum number of SSH authentication attempts. Range: 1 to 10. Default: 6.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Setting the maximum number of authentication attempts:

```
switch(config)# ssh maximum-auth-attempts 3
```

Resetting the maximum number of authentication attempts to its default of 6:

```
switch(config)# no maximum-auth-attempts
```

## ssh public-key-algorithms

### Syntax

```
ssh public-key-algorithms <PUBLIC-KEY-ALGORITHMS-LIST>  
no ssh public-key-algorithms
```

### Description

Configures SSH to use a set of public key algorithms in the specified priority order. The first public key type entered in the CLI is considered a first priority. Public key algorithms specify which public key types can be used for public key authentication in SSH. Each option represents a public key type that the SSH server can accept or that the SSH client can present to a server. Only the public key algorithms that are chosen by the user are configured.

The `no` form of this command removes the configuration of public key algorithms and reverts SSH to use the default set.

## Command context

config

## Parameters

<PUBLIC-KEY-ALGORITHMS-LIST>

Valid public key algorithm types are:

- ecdsa-sha2-nistp256
- ecdsa-sha2-nistp384
- ecdsa-sha2-nistp521
- ssh-ed25519
- ssh-rsa
- rsa-sha2-256
- rsa-sha2-512
- x509v3-ecdsa-sha2-nistp256
- x509v3-ecdsa-sha2-nistp384
- x509v3-ecdsa-sha2-nistp521
- x509v3-rsa2048-sha256
- x509v3-sign-rsa
- x509v3-ssh-rsa

Default set of public key algorithms in priority order:

1. rsa-sha2-256
2. rsa-sha2-512
3. ssh-rsa
4. ecdsa-sha2-nistp256
5. ecdsa-sha2-nistp384
6. ecdsa-sha2-nistp521
7. ssh-ed25519
8. x509v3-rsa2048-sha256
9. x509v3-ssh-rsa
10. x509v3-sign-rsa
11. x509v3-ecdsa-sha2-nistp256
12. x509v3-ecdsa-sha2-nistp384
13. x509v3-ecdsa-sha2-nistp521

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring SSH to use a set of specified public key algorithms:

```
switch(config)# ssh public-key-algorithms x509v3-ssh-rsa ssh-rsa rsa-sha2-256
```

Reverting SSH to use the default set of public key algorithms:

```
switch(config)# no ssh public-key-algorithms
```

## ssh server vrf

### Syntax

```
ssh server vrf <VRF-NAME>  
no ssh server vrf <VRF-NAME>
```

### Description

Enables the SSH server on the specified VRF.

The `no` form of the command disables the SSH server on the specified VRF.

### Command context

config

### Parameters

vrf <VRF-NAME>

Specifies the VRF name.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Enabling the SSH server on the management VRF:

```
switch(config)# ssh server vrf mgmt
```

Disabling the SSH server on the management VRF:

```
switch(config)# no ssh server vrf mgmt
```

The switch provides an SSH client that enables the switch to log in to an SSH server such as another switch, typically for command execution purposes. The SSH client provides secure encrypted communications between the switch and the SSH server over any network.

## SSH client commands

### ssh (client login)

#### Syntax

```
ssh [<USERNAME>@]{<IPV4> | <HOSTNAME>} [vrf <VRF-NAME>] [port <PORT-NUMBER>]
```

#### Description

Establishes a client session with an SSH server which is typically another switch.

#### Command context

Manager (#)

#### Parameters

<USERNAME>

Specifies the username that the client uses to log in to an SSH server. When omitted, the username of the current session is used.

{<IPV4> | <HOSTNAME>}

Specifies the SSH server to which the SSH client will connect.

- <IPV4>: The IPv4 address.
- <HOSTNAME>: The host name.

vrf <VRF-NAME>

Specifies the VRF to be used for the SSH client session. When omitted, the default VRF named `default` is used.

port <PORT-NUMBER>

Specifies the SSH server TCP port number. When omitted, the default TCP port 22 is used.

#### Authority

Administrators or local user group members with execution rights for this command.

#### Examples

Establishing an SSH client session (using the management VRF) with an SSH server:

```
switch# ssh admin@10.0.11.180 vrf mgmt
```

Establishing an SSH client session (using the default VRF and a specific port) with an SSH server:

```
switch# ssh admin@10.0.11.175 port 223
```

Configuring a test user on switch 1 and then connecting to switch 1 from switch 2 using the SSH client on the mgmt VRF:

```
** Configuring a test user on switch 1 **  
switch(config)# user-group test  
switch(config-usr-grp-test)# permit cli command ".*"  
switch(config)# exit  
switch(config)# user test-user group test password plaintext tst#9J%** On switch 2,  
connecting to switch 1 using the SSH client **  
switch# ssh test-user@10.0.11.177 vrf mgmt
```



Local AAA on your Aruba switch provides:

- Authentication using local password or SSH public key.
- Authorization using local role-based access control (RBAC). Optional per-command authorization is possible through configuration of user-defined local user groups, with command authorization rules applied to respective group members.
- Accounting of user activity on the switch using accounting logs.



For switches that support multiple management modules such as the Aruba 8400, all AAA functionality discussed only applies to the active management module. See also *AAA on switches with multiple management modules* in the *High Availability Guide*.

## Local AAA defaults and limits

Setting	Default value / limit
Local authentication	Enabled by default for all connection types: console, SSH, and REST.
Local role-based access control (RBAC) authorization	Enabled by default for all connection types: console, SSH, and REST.
Local accounting	Enabled.
Maximum number of local users	64 users, including the default <code>admin</code> user.
Maximum number of user-defined local user groups	32 groups, including the three built-in groups <code>administrators</code> , <code>operators</code> , <code>auditors</code> .
Password for default admin account	The password is empty by default.
SSH public key authentication	Enabled.

## Local authentication

Authentication identifies users, validates their credentials, and grants switch access. Local authentication is either password-based or SSH public key-based.

### Password-based local authentication

- Validates users with local user name and password credentials
- Is supported on all interfaces/channels (SSH, WebUI, Console, REST)

- Is enabled by default but can be superseded by remote authentication or with SSH client using SSH public key authentication

## SSH public key-based local authentication

- Validates users identified with SSH public keys stored in the local user database
- Is supported on the SSH interface/channel with SSH client
- Takes precedence over password-based authentication whether local or remote
- Is enabled by default (also requires key configuration to work)

## Local authentication tasks

The local authentication (local password and SSH public key) tasks are as follows:

Task	Command name	Example
Enable authentication as local for the specified connection types	aaa authentication login	Enable local authentication for the default and console connection types: aaa authentication login default local aaa authentication login console local
Show authentication configuration	show aaa authentication	show aaa authentication
Enable password-based authentication minimum password length checking	aaa authentication minimum-password-length	aaa authentication minimum-password-length 12
Disable password-based authentication minimum password length checking	aaa authentication minimum-password-length	no aaa authentication minimum-password-length

Task	Command name	Example
Enable local password-based authentication login attempt limiting	aaa authentication limit-login-attempts	aaa authentication limit-login-attempts 4 logout-time 20
Disable local password-based authentication login attempt limiting	aaa authentication limit-login-attempts	no aaa authentication limit-login-attempts
Enable local password-based authentication for use with SSH clients (enabled by default)	ssh password-authentication	ssh password-authentication
Disable local password-based authentication for use with SSH clients	ssh password-authentication	no ssh password-authentication
Enable SSH public key authentication (enabled by default)	ssh public-key-authentication	ssh public-key-authentication
Disable SSH public key authentication	ssh public-key-authentication	no ssh public-key-authentication

Task	Command name	Example
Show state of local password-based (for SSH) and SSH public key authentication	<code>show ssh authentication-method</code>	<code>show ssh authentication-method</code>
Copying the client SSH public key into the key list	<code>user authorized-key</code>	<code>user admin authorized-key ecdsa-sha2-nistp256 E2VjZH...QUiCAk=root@switch</code>
Removing SSH public keys from the key list	<code>user authorized-key</code>	<code>no user admin authorized-key 2</code>
Showing the SSH client public key list	<code>show user</code>	<code>show user admin authorized-key</code>

## Local authorization

Authorization controls authenticated users command execution and switch interaction privileges. Local authorization uses role-based access control (RBAC) to provide role-based privilege levels plus optional user-defined local user groups with command execution rules. Authorization occurs only after successful authentication.

- **Administrators** have full command execution and switch interaction privilege.
- **Operators** are limited to the use of several nonsensitive `show` commands.
- **Auditors** are limited to a few auditing-related commands.

Optional per-command authorization is available through configuration of user-defined local user groups with command authorization rules applied to respective group members. see [User-defined user groups](#) .

## Local authorization tasks

The local authorization tasks are as follows:

Task	Command name	Example
Enable authorization as local RBAC for the specified connection types	<code>aaa authorization commands</code>	Enable local authorization for the default and console connection types: <code>aaa authorization commands default local</code> <code>aaa authorization commands console local</code>
Show authorization configuration	<code>show aaa authorization</code>	<code>show aaa authorization</code>

## Local accounting

Local accounting is always active. It cannot be turned off.

This accounting information is captured and made available locally (using `show accounting log`) and, if desired, for sending to remote AAA servers:

- Exec Accounting: user login/logout events.
- Command accounting: commands executed by users.
- System accounting: remote accounting On/Off events.
- CLI show commands.
- Interactions on the non-CLI interfaces: REST and WebUI.

The following is not captured or made available as accounting information:

- CLI commands that reboot the switch.
- Interactions in the bash shell.



---

See also the `show accounting log` command.

---

## Local accounting tasks

The local accounting tasks are as follows:

Task	Command name	Example
Enable accounting as local for the specified connection types	<code>aaa accounting all-mgmt</code>	Enable local accounting for the default and console connection types: <code>aaa accounting all-mgmt default start-stop local</code> <code>aaa accounting all-mgmt console start-stop local</code>
Show accounting configuration	<code>show aaa accounting</code>	<code>show aaa accounting</code>
Show local accounting log contents	<code>show accounting log</code>	<code>show accounting log last 10</code>

### aaa accounting all-mgmt

#### Syntax

```
aaa accounting all-mgmt <CONNECTION-TYPE> start-stop {local | group <GROUP-LIST>}  
no aaa accounting all-mgmt <CONNECTION-TYPE>
```

#### Description

Defines accounting as being local (with the name `local`) (the default). Or defines a sequence of remote AAA server groups to be accessed for accounting purposes.

For remote accounting, the information is sent to the first reachable remote server that was configured with this command for remote accounting. If no remote server is reachable, local accounting remains available. Each available connection type (channel) can be configured individually as either local or using remote AAA server groups. All server groups named in your command, must exist. This command can be issued multiple times, once for each connection type. Local is always available for any connection type not configured for remote accounting.



---

The system accounting log is not associated with any connection type (channel) and is therefore sent to the accounting method configured on the default connection type (channel) only.

---

The `no` form of this command removes for the specified connection type, any defined remote AAA server group accounting sequence. Local accounting is available for connection types without a configured remote AAA server group list (whether default or for the specific connection type).

#### Command context

`config`

#### Parameters

`<CONNECTION-TYPE>`

One of these connection types (channels):

`default`

Defines a list of accounting server groups to be used for the `default` connection type. This configuration applies to all other connection types (`console`, `https-server`, `ssh`) that are not explicitly configured with this command. For example, if you do not use `aaa accounting all-mgmt console...` to define the console accounting list, then this default configuration is used for console.

`console`

Defines a list of accounting server groups to be used for the `console` connection type.

`https-server`

Defines a list of accounting server groups to be used for the `https-server` (REST, Web UI) connection type.

`ssh`

Defines a list of accounting server groups to be used for the `ssh` connection type.

`start-stop`

Selects accounting information capture at both the beginning and end of a process.

`local`

Selects local-only accounting when used without the `group` parameter.

`group <GROUP-LIST>`

Specifies the list of remote AAA server group names. Each name can be specified one time. Predefined remote AAA group names `tacacs` and `radius` are available. Although not a group name, predefined name `local` is available. User-defined TACACS+ and RADIUS server group names may also be used. The remote AAA server groups are accessed in the order that the group names are listed in this command. Within each group, the servers are accessed in the order in which the servers were added to the group. Server groups are defined using command `aaa group server` and servers are added to a server group with the command `server`.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

Local accounting is always active. It cannot be turned off.

## Examples

Setting local accounting for the default connection type:

```
switch(config)# aaa accounting all-mgmt default start-stop local
```

Setting local accounting for the console connection type:

```
switch(config)# aaa accounting all-mgmt console start-stop local
```

# aaa authentication console-login-attempts

## Syntax

```
aaa authentication console-login-attempts <ATTEMPTS> console-lockout-time <LOCKOUT-TIME>  
no aaa authentication console-login-attempts
```

## Description

For the console interface only (not SSH or REST), enables console login attempt limiting. If the number of failed console login attempts equals the configured threshold, the user is locked out for the configured duration..

The `no` form of this command disables console login attempt limits.



**Important:** If you enable the lockout using this command and also enable the SSH and REST lockout using command `aaa authentication limit-login-attempts`, and then enter too many consecutive wrong passwords, you will become locked out, and will have to wait for the configured lockout time to elapse before logging in on any interface.



---

This console login attempt limiting feature is only available when not using remote authentication through AAA servers (TACACS+ or RADIUS) on any interface. Remote authentication through AAA servers (TACACS+ or RADIUS) is not possible when limit login attempts is configured on any interface.

---

## Command context

config

## Parameters

*<ATTEMPTS>*

Specifies the threshold of failed console login attempts that triggers user lockout. Range: 1 to 10. For example, if *<ATTEMPTS>* is set to 1, a single failed login attempt triggers immediate user lockout.

*<LOCKOUT-TIME>*

Specifies the amount of time a user is locked out. Range: 1 to 3600 seconds.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Enabling console login attempt failure limiting with a 60 second lockout being triggered upon the third consecutive login attempt failure.

```
switch(config)# aaa authentication console-login-attempts 3 console-lockout-time 60
```

Disabling console login attempt failure limiting:

```
switch(config)# no aaa authentication console-login-attempts
```

# aaa authentication limit-login-attempts

## Syntax

```
aaa authentication limit-login-attempts <ATTEMPTS> lockout-time <LOCKOUT-TIME>  
no aaa authentication limit-login-attempts
```

## Description

For the SSH and REST interface, enables local login attempt limiting. If the number of failed local login attempts equals the configured threshold, the user is locked out for the configured duration.

The `no` form of this command disables local login attempt limits.



---

**Important:** If you enable the lockout using this command and also enable the console lockout using command `aaa authentication console-login-attempts`, and then enter too many consecutive wrong passwords, you will become locked out, and will have to wait for the configured lockout time to elapse before logging in on any interface.

---





---

This local login attempt limiting feature is only available when not using remote authentication through AAA servers (TACACS+ or RADIUS) on any interface. Remote authentication through AAA servers (TACACS+ or RADIUS) is not possible when limit login attempts is configured on any interface.

---

## Command context

config

## Parameters

<ATTEMPTS>

Specifies the threshold of failed local login attempts that triggers user lockout. Range: 1 to 10. For example, if <ATTEMPTS> is set to 1, a single failed login attempt triggers immediate user lockout.

<LOCKOUT-TIME>

Specifies the amount of time a user is locked out. Range: 1 to 3600 seconds.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Enabling local login attempt failure limiting with a 20 second lockout being triggered upon the fourth consecutive login attempt failure.

```
switch(config)# aaa authentication limit-login-attempts 4 lockout-time 20
```

Disabling login attempt failure limiting:

```
switch(config)# no aaa authentication limit-login-attempts
```

# aaa authentication login

## Syntax

```
aaa authentication login <CONNECTION-TYPE> {local | group <GROUP-LIST>}  
no aaa authentication login <CONNECTION-TYPE>
```

## Description

Defines authentication as being local (with the name `local`) (the default). Or defines a sequence of remote AAA server groups to be accessed for authentication purposes. Each available connection type (channel) can be configured individually as either local or using remote AAA server groups. All server groups named in your command, must exist. This command can be issued multiple times, once for each connection type. Local is always available for any connection type not configured for remote AAA authentication.

The `no` form of this command removes for the specified connection type, any defined remote AAA server group authentication sequence. Local authentication is available for connection types without a configured remote AAA server group list (whether default or for the specific connection type).

## Command context

config

## Parameters

<CONNECTION-TYPE>

One of these connection types (channels):

`default`

Defines a list of authentication server groups to be used for the `default` connection type. This configuration applies to all other connection types (`console`, `https-server`, `ssh`) that are not explicitly configured with this command. For example, if you do not use `aaa authentication login console...` to define the console authentication list, then this default configuration is used for `console`.

`console`

Defines a list of authentication server groups to be used for the `console` connection type.

`https-server`

Defines a list of authentication server groups to be used for the `https-server` (REST, Web UI) connection type.

`ssh`

Defines a list of authentication server groups to be used for the `ssh` connection type.

`local`

Selects local-only authentication when used without the `group` parameter.

`group` <GROUP-LIST>

Specifies the list of remote AAA server group names. Each name can be specified one time. Predefined remote AAA group names `tacacs` and `radius` are available. Although not a group name, predefined name `local` is available. User-defined TACACS+ and RADIUS server group names may also be used. The remote AAA server groups are accessed in the order that the group names are listed in this command. Within each group, the servers are accessed in the order in which the servers were added to the group. Server groups are defined using command `aaa group server` and servers are added to a server group with the command `server`.

If no AAA server is reachable, local authentication is attempted.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Setting local authentication for the default connection type:

```
switch(config)# aaa authentication login default local
```

Setting local authentication for the console connection type:

```
switch(config)# aaa authentication login console local
```

## aaa authentication minimum-password-length

### Syntax

```
aaa authentication minimum-password-length <LENGTH>  
no aaa authentication minimum-password-length
```

### Description

Enables minimum password length checking. Existing passwords shorter than the minimum length are unaffected. Length checking does not apply to ciphertext passwords. Length checking applies both to local and remote authentication.

The `no` form of this command disables minimum password length checking.

## Command context

config

## Parameters

<LENGTH>

Specifies the minimum password length. Range: 1 to 32.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Enabling password length checking, with a minimum length of 12.

```
switch(config)# aaa authentication minimum-password-length 12
```

Disabling minimum password length checking:

```
switch(config)# no aaa authentication minimum-password-length
```

# aaa authorization commands

## Syntax

```
aaa authorization commands <CONNECTION-TYPE> {local | none}
aaa authorization commands <CONNECTION-TYPE> group <GROUP-LIST>

no aaa authorization commands <CONNECTION-TYPE>
```

## Description

Defines authorization as being basic local RBAC (specified as `none`), or as full-fledged local RBAC specified as `local` (the default), or as remote TACACS+ (specified with `group <GROUP-LIST>`). Each available connection type (channel) can be configured individually. All server groups named in the command, must exist. This command can be issued multiple times, once for each connection type.

The `no` form of this command unconfigures authorization for the specified connection type, reverting to the default of `local`.



---

Although only TACACS+ servers are supported for remote authorization, local authorization (basic or full-fledged) can be used with remote RADIUS authentication.

---

## Command context

config

## Parameters

<CONNECTION-TYPE>

One of these connection types (channels):

`default`

Selects the `default` connection type for configuration. This configuration applies to all other connection types (`console`, `ssh`) that are not explicitly configured with this command. For example, if you do not use `aaa authorization commands console...` to define the console authorization list, then this default configuration is used for console.

`console`

Selects the `console` connection type for configuration.

`ssh`

Selects the `ssh` connection type for configuration.

`local` (the default)

When used alone without `group <GROUP-LIST>`, selects local authorization which can be used to provide authorization for a purely local setup without any remote AAA servers and also for when RADIUS is used for remote Authentication and Accounting but Authorization is local.

When used after `group`, provides for fallback (to full-fledged local authorization) when every server in every specified TACACS+ server group cannot be reached.



---

If any TACACS+ server in the specified groups is reachable, but the command fails to be authorized by that server, the command is rejected and local authorization is never attempted. Local authorization is only attempted if every TACACS+ server cannot be reached.

---

`none`

When used alone without `group <GROUP-LIST>`, selects basic local RBAC authorization, for use with the built-in user groups (`administrators`, `operators`, `auditors`).

When used after `group`, provides for fallback (to basic local RBAC authorization) when every server in every specified TACACS+ server group cannot be reached.



---

With `none`, for users belonging to user-defined user groups, all commands can be executed regardless of what authorization rules are defined in such groups. For per-command local authorization, use `local` instead.

---

`group <GROUP-LIST>`

Specifies the list of remote AAA server group names. Predefined remote AAA group name `tacacs` is available. User-defined TACACS+ server group names may also be used. The remote AAA server groups are accessed in the order that the group names are listed in this command. Within each group, the servers are accessed in the order in which the servers were added to the group. Server groups are defined using command `aaa server group` and servers are added to a server group using command `server`.

It is recommended to always include either the special name `local` or `none` as the last name in the group list. If both `local` and `none` are omitted, and no remote AAA server is reachable (or the first reachable server cannot authorize the command), command execution for the current user will not be possible.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Setting the authorization for default to `local`:

```
switch(config)# aaa authorization commands default local
```

Setting the authorization for the SSH interface to `none`:

```
switch(config)# aaa authorization commands ssh none
```

## show aaa accounting

### Syntax

```
show aaa accounting [vsx-peer]
```

### Description

Shows the accounting configuration per connection type (channel).

### Command context

Operator (>) or Manager (#)

### Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Example

Configuring and then showing local accounting for the default and console connection types:

```
switch(config)# aaa accounting all default start-stop local
switch(config)# aaa accounting all console start-stop local
switch(config)# exit
switch# show aaa accounting
AAA Accounting:
  Accounting Type           : all
  Accounting Mode           : start-stop
```

Accounting for default channel:

GROUP NAME	GROUP PRIORITY
local	0

Accounting for console channel:

GROUP NAME	GROUP PRIORITY
local	0

## show aaa authentication

### Syntax

```
show aaa authentication [vsx-peer]
```

### Description

Shows the authentication configuration per connection type (channel).

## Command context

Operator (>) or Manager (#)

## Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

## Example

Configuring and then showing local authentication for the default and console connection types (channels):

```
switch(config)# aaa authentication login default local
switch(config)# aaa authentication login console local
switch(config)# exit
switch# show aaa authentication
```

```
AAA Authentication:
  Fail-through           : Disabled
  Limit Login Attempts   : Not set
  Lockout Time           : 300
  Minimum Password Length : Not set
```

```
Authentication for default channel:
```

```
-----
GROUP NAME | GROUP PRIORITY
-----
local      | 0
-----
```

```
Authentication for console channel:
```

```
-----
GROUP NAME | GROUP PRIORITY
-----
local      | 0
-----
```

## show aaa authorization

### Syntax

```
show aaa authorization [vsx-peer]
```

### Description

Shows the authorization configuration per connection type (channel).

## Command context

Operator (>) or Manager (#)

## Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Example

Configuring and then showing full-fledged local RBAC authorization for the default and console connection types (channels):

```
switch(config)# aaa authorization commands default none
switch(config)#
switch(config)# aaa authorization commands console none
switch(config)# exit
switch#
switch# show aaa authorization
Authorization for default channel:
```

GROUP NAME	GROUP PRIORITY
none	0

Authorization for console channel:

GROUP NAME	GROUP PRIORITY
none	0

## show ssh authentication-method

### Syntax

```
show ssh authentication-method
```

### Description

Shows the status of the SSH public key method and the local password-based (through SSH client) authentication method.

### Command context

Operator (>) or Manager (#)

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Example

Showing the authentication methods.

```
switch# show ssh authentication-method
SSH publickey authentication : Enabled
SSH password authentication : Enabled
```

## show user

### Syntax

```
show user <USERNAME> authorized-key
```

### Description

Shows the SSH client public key list for a specified user.

### Command context

Operator (>) or Manager (#)

### Parameters

<USERNAME>

Specifies the username for which you want to show the SSH client public key list.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Usage

Any user can show their own public key list; however, administrators can also show a public key list of other users.

### Examples

Showing a client public key:

```
switch# show user admin authorized-key

1. Key Type : RSA      Key size : 2048
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDMtyMBmmAaF6r1zxf3DZNHSYVHBjhlbBlyAIqQ8DSHK
...
U+aEl4UW/ifIukmK67sIHwK+FhhRYwPztQc5pjyOPk128a4pgKQaHCcOF169Z admin@switch
```

Showing two client public keys:

```
switch# show user admin authorized-key

1. Key Type : ECDSA      Curve : nistp256
ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBEqEFevZ0
...
176V+D0svdCJ9Wo32zqI9OeAdTJw/eZYp5qknhNgS81HjAI6J/4/kAqdZAJbqQUiCAk= admin@switch

2. Key Type : RSA      Key size : 2048
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQDXQHrQV7+/GcMdOhr//IRjJkX7TQKupW89j80bL7xq8
...
j8qKuHWSN0/h/HxjzQJuYDVmZN5vG3DhpXbBZU1ZNnchVod13QLCesqA3VLKN admin@switch
```



## ssh password-authentication

### Syntax

```
ssh password-authentication
no ssh password-authentication
```

### Description

Enables the password-based authentication method for use with SSH clients.

The `no` form of this command disables the password-based authentication method for use with SSH clients.

### Command context

config

### Authority

Administrators or local user group members with execution rights for this command.

### Usage

The switch ships with password-based authentication (for SSH clients) enabled. The maximum number of password retries is three.

### Examples

Enabling password authentication for use with SSH clients:

```
switch(config)# ssh password-authentication
```

Disabling password authentication for use with SSH clients:

```
switch(config)# no ssh password-authentication
```

## ssh public-key-authentication

### Syntax

```
ssh public-key-authentication
no ssh public-key-authentication
```

### Description

Enables the SSH public key authentication method. The switch ships with SSH public key authentication enabled.

The `no` form of this command disables the SSH public key authentication method.



---

Although SSH public key authentication is enabled by default, it cannot be used until SSH public keys are added with the `user authorized-key` command.

---

### Command context

config

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Enabling SSH public key authentication:

```
switch(config)# ssh public-key-authentication
```

Disabling SSH public key authentication:

```
switch(config)# no ssh public-key-authentication
```

# user authorized-key

## Syntax

```
user <USERNAME> authorized-key <PUBKEY>  
no user <USERNAME> authorized-key [<KEYNUM>]
```

## Description

Copies an SSH client public key into the key list. If the key list and the public key do not exist, it creates a list with the public key. If the SSH client public key exists, the command appends the new key to the existing list. The client public key list holds a maximum of 32 client keys.

The `no` form of the command removes either one or all SSH public keys from the key list.

## Command context

config

## Parameters

<USERNAME>

Specifies the name of the user.

<PUBKEY>

Specifies the SSH client public key to be copied into the key list.

<KEYNUM>

Specifies the key number. The range is 1 to 32. Use the `show user <USERNAME> authorized-key` command to find the key number associated with the key.

## Authority

Operators or Administrators or local user group members with execution rights for this command.

Operators can execute this command from the operator context (>) only.

## Usage

Each key on the key list has a key identifier. The `show user <USERNAME> authorized-key` command displays the key identifier associated with the key.

Administrators can add and remove the public keys of themselves and other users. Operators can add and remove only their own public keys. If the public key authentication method is enabled, the client public key present is used by the SSH server to authenticate the client. The authentication method reverts to the password authentication method and prompts for a client password when one of the following occurs:

- The client public keys are not present.
- The server does not have the keys enabled.
- The public key method is disabled.

You can either remove all keys or a specific key. Each key on the key list has a key identifier. If you provide the key identifier in this command, the command removes the corresponding key from the list. If you provide no key identifier, the command removes all keys from the key list.

## Examples

Adding a public key:

```
switch(config)#user admin authorized-key ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBEqEFevZ0176V+D0svdCJ9Wo32zqI9OeAIdTJwT/eZYp50qkA
nhZNgS81HBjAI6QJ/4/kAyqdZ9oAjbiqQUiCAk= root@switch
```

Removing all SSH public keys from the list:

```
switch(config)# no user admin authorized-key
```

Removing the specified SSH public key from the list:

```
switch(config)# no user admin authorized-key 2
```

Remote AAA provides the following for your Aruba switch:

- Authentication using remote TACACS+ AAA servers.
- Authorization using remote TACACS+ AAA servers, providing fine-grained command authorization. Optional user-defined local user groups with configured command authorization rules can be used to provide authorization fallback protection for when TACACS+ servers become temporarily unavailable.
- Transmission of locally collected accounting information to remote TACACS+ servers.



For switches that support multiple management modules such as the Aruba 8400, all AAA functionality discussed only applies to the active management module. See also *AAA on switches with multiple management modules* in the *High Availability Guide*.

## Default server groups

The switch always has these four default groups:

- `tacacs`: for remote AAA, always contains every configured TACACS+ server.
- `radius`: for remote AAA, always contains every configured RADIUS server.
- `local`: for local authentication.
- `none`: for local (RBAC) authorization.

User-defined AAA servers are always added to the matching default group, either `tacacs` or `radius`. Optionally, each server can also be added to exactly one additional user-defined (custom) group. A maximum of 28 user-defined groups can be created.

The order in which servers are added to a group is important. The server added first is accessed first, and if necessary, the second server is accessed second, and so on.

## Remote AAA (TACACS+) defaults and limits

Setting	Default value / limit
Authentication of REST sessions with TACACS+	Disabled
Maximum number of TACACS+ servers in an AAA group	16
Maximum number of TACACS+ servers that can be configured	16
Maximum number of user-defined AAA server groups that can be configured	28
TACACS+ authentication	Disabled

Setting	Default value / limit
TACACS+ authentication global timeout	5 seconds
TACACS+ authentication passkey (shared secret)	None
TACACS+ authentication tcp-port	49
TACACS+ global authentication protocol	PAP
TACACS+ server tracking default interval	300 seconds
TACACS+ server access through the default VRF	default*

\*The default value is `default`, unless another VRF is specified during the server configuration.

## About global versus per-TACACS+ server passkeys (shared secrets)

To communicate with a TACACS+ AAA server, the switch must have a passkey (shared secret) configured that matches what is configured on the server. Use one of these commands to achieve the desired configuration:

- For a global passkey common to every TACACS+ server, use `tacacs-server key`.
- For a per-TACACS+ server passkey, use `tacacs-server host` with the `key` parameter.




---

If both passkeys are configured on the switch, the per-TACACS+ server passkey is used.

---

## Remote AAA TACACS+ server configuration requirements

The user-supplied TACACS+ server must:

- Have an IPv4/IPv6 address or fully qualified domain name (FQDN) that is visible to the switch.
- Have a passkey (shared secret) that matches what is configured on the switch.
- Provide username and password definitions for every switch user. Remote users do not require definition on the switch.
- Configure user role assignment using TACACS+ attributes.
- Have any needed command authorization configured to control what commands (per user or user role) will be executable on the switch.




---

Consult your TACACS+ server documentation for installation and general configuration details.

---




---

If SSH public key authentication is used, the key information is stored locally on the switch, making username and password definition on the TACACS+ server unnecessary.

---

## User role assignment using TACACS+ attributes

User role assignment is configured on the TACACS+ server using VSAs (vendor-specific attributes) and TACACS+ specified attributes.

TACACS+ servers can return multiple attribute value pairs (AVPs) in response to an authentication request. The attributes are processed in this order of precedence to determine the user role assigned:

- If the `Aruba-Admin-Role` VSA is present, map the user to the matching corresponding local user-group name.
  - Else if the `priv-lvl` TACACS+ specified attribute is present, extract the privilege level (1, 15, or 19) and map the user to the local user-group corresponding to this privilege level (1=operators, 15=administrators, 19=auditors). Privilege levels 2 to 14 may also be used with matching local user groups named 2 to 14.
  - Otherwise, the user role cannot be determined, and authentication fails.

Aruba-Admin-Role	priv-lvl	User role assigned
<GROUP-NAME>	Do not care	Matching local user <GROUP-NAME>
Not present	1	Operators
Not present	15	Administrators
Not present	19	Auditors
Not present	2 to 14	Matching local user groups named 2 to 14
Not present	Not present	None (not authenticated)

## TACACS+ server redundancy and access sequence

To prevent authentication and authorization interruption, it is common practice to configure more than one TACACS+ server. When identifying TACACS+ servers to the switch, server group order (and server order within the group), determines server access order.



When defining the server access sequence for authentication with `aaa authentication login default`, there is an implied `local` included as the last item in the list. If no TACACS+ server can be reached, local authentication will be attempted.



When defining the server access sequence for authorization with `aaa authorization commands`, it is recommended to always include either `local` or `none` as the last item in the list.

## Single source IP address for consistent source identification to AAA servers



If applicable to your installation, it is recommended that you perform the optional configuration mentioned in this section.

If your topology allows the AAA server to be reached through multiple paths, the server interprets the incoming packets to be from different switches even though they are all coming from the same switch.

Having a switch associated with multiple IP addresses makes it more difficult to interpret system logs and accounting data.

To ensure that all traffic sent from the switch to the AAA server uses the same source IP address, use `ip source-interface` or `ipv6 source-interface`. These two commands plus the related commands `show ip source-interface` and `show ipv6 source-interface` are described under *Layer 2/3 Interface commands* in the *Command-Line Interface Guide*.

## TACACS+ general tasks

General TACACS+ tasks, not specific to authentication, authorization, or accounting, are as follows:

Task	Command name	Example
Configuring a TACACS+ server	<code>tacacs-server host</code>	<code>tacacs-server host 1.1.1.1 vrf default</code> <code>no tacacs-server host 1.1.1.1 vrf default</code>
Showing global and TACACS+ server configurations	<code>show tacacs-server</code>	<code>show tacacs-server detail</code>
Configuring a TACACS+ server group	<code>aaa group server</code>	<code>aaa group server tacacs sgl</code> <code>no aaa group server tacacs sgl</code>
Showing server groups	<code>show aaa server-groups</code>	<code>show aaa server-groups</code>
Adding a TACACS server to a server-group	<code>server</code>	<code>aaa group server tacacs sgl</code> <code>server 1.1.1.2 port 32 vrf default</code>
Deleting a TACACS server from a server-group	<code>server</code>	<code>aaa group server tacacs sgl</code> <code>no server 1.1.1.2 port 32 vrf default</code>
Configuring a TACACS+ global passkey	<code>tacacs-server key</code>	<code>tacacs-server key plaintext mypasskey123</code>
Configuring PAP or CHAP for TACACS+	<code>tacacs-server auth-type</code>	<code>tacacs-server auth-type chap</code> <code>no tacacs-server auth-type</code>
Configuring the TACACS+ global timeout	<code>tacacs-server timeout</code>	<code>tacacs-server timeout 20</code> <code>no tacacs-server timeout</code>

## TACACS+ authentication

TACACS+ authentication occurs as follows:

- User credentials are sent from the switch to TACACS+ server using the PAP or CHAP authentication protocol.
- If a user is authenticated, their role is communicated to the switch as Administrator, Operator, or Auditor.
- An unknown user or a user who entered an invalid password is identified as such to the switch, which then rejects user login.

## About authentication fail-through

Normally, authentication is performed by the first AAA server reached. A rarely needed feature named "Authentication fail-through" is available. If Authentication fail-through is enabled and authentication fails on the first reachable AAA server, authentication is attempted on the second AAA server, and so on, until successful authentication or the server list is exhausted.

Enabling Authentication fail-through is typically unnecessary because the user credential databases should be consistent across all AAA servers. Authentication fail-through might be helpful if your AAA user credential databases are not quickly synchronized across all AAA servers.

## TACACS+ authentication tasks

The TACACS+ authentication-related tasks are as follows:

Task	Command name	Example
Configuring the authentication sequence for the default connection type	<code>aaa authentication login</code>	<code>aaa authentication login default group tg1 tg2 tacacs local</code>
Configuring the authentication sequence for the console connection type	<code>aaa authentication login</code>	<code>aaa authentication login console group tg2 tg3 tacacs local</code>
Configuring the authentication sequence for the ssh connection type	<code>aaa authentication login</code>	<code>aaa authentication login ssh group tg2 tacacs local</code>
Removing remote AAA for the default connection type	<code>aaa authentication login</code>	<code>no aaa authentication login default</code>
Configuring authentication fail-through	<code>aaa authentication allow-fail-through</code>	<code>aaa authentication allow-fail-through</code> <code>no aaa authentication allow-fail-through</code>
Showing the authentication sequence	<code>show aaa authentication</code>	<code>show aaa authentication</code>

## TACACS+ authorization

Upon successful user authentication, the user is assigned their role by the TACACS+ server. See also [User role assignment using TACACS+ attributes](#).



TACACS+ authorization provides command filtering to allow/disallow individual command or command set execution. Each command is sent to the TACACS+ server for approval, and the switch then allows/disallows command execution according to the server response.



---

TACACS+ authorization applies only to the CLI interface.

---

## Using local authorization as fallback from TACACS+ authorization

Local authorization can be used for the situation in which communication is lost with all TACACS+ servers after a successful authentication. Users that are members of the built-in local user groups (*administrators*, *operators*, or *auditors*) are authorized according to the fixed roles and privilege levels of those groups. Optionally, local user-defined user groups can be configured with specific command execution rules per group. Users that are members of such groups, are authorized according to the command execution rules of the group to which they belong. For configuring local user groups, see *user-group*.

## About authentication fail-through and authorization

For authorization, there is no equivalent of the authentication fail-through feature. Therefore, if the first reachable TACACS+ server responds with "Authorization Denied," no additional TACACS+ servers are interrogated.



---

Rare potential out-of-synchronization situation when using authentication fail-through: Successful authentication on one server can be followed by authorization denial on another. The user is known on the server doing the authentication but unknown on the server attempting the authorization. This situation typically arises only during brief periods in which user credential databases are not synchronized across all TACACS+ servers. See also TACACS+ server authorization considerations in [aaa authorization commands](#).

---

## TACACS+ authorization tasks

The TACACS+ authorization-related tasks are as follows:

Task	Command name	Example
Configuring the authorization sequence for the default connection type	<code>aaa authorization commands</code>	<code>aaa authorization commands default group tg1 tacacs local</code>
Configuring the authorization sequence for the console connection type	<code>aaa authorization commands</code>	<code>aaa authorization commands console group tg1 tg2 tacacs none</code>

Task	Command name	Example
Removing remote AAA for the default connection type	aaa authorization commands	no aaa authorization commands default
Showing the TACACS+ authorization sequence	show aaa authorization	show aaa authorization

## TACACS+ accounting

This accounting information is captured and made available for sending to remote accounting servers:

- Exec Accounting: user login/logout events.
- Command accounting: commands executed by users.
- System accounting: remote accounting On/Off events.
- CLI show commands.
- Interactions on the non-CLI interfaces: REST and WebUI.

The following is not captured or made available as accounting information:

- CLI commands that reboot the switch.
- Interactions in the bash shell.




---

Local accounting (always enabled) must be functioning properly for remote Accounting to work.

---




---

The accounting information is sent to the first reachable remote TACACS+ AAA server (configured for remote accounting). If no remote TACACS+ server is reachable, local accounting remains available.

---

## Sample accounting information on a TACACS+ server

```
Mon May 9 17:52:32 10.10.11.1 UNKNOWN tty 0.0.0.0 start task_id=1525899775430
timezone=UTC start_time=1525913552.428 service=system event=sys_acct
reason="System-accounting-ON" result=success
Mon May 9 17:52:48 10.10.11.1 admin tty 192.168.1.20 start task_id=1525899775431
timezone=UTC start_time=1525913567.611 service=shell priv_lvl=15 result=success
Mon May 9 17:52:48 10.10.11.1 admin tty 192.168.1.20 stop task_id=1525899775432
timezone=UTC stop_time=1525913567.614 service=shell priv_lvl=15 cmd="enable"
result=success
Mon May 9 17:52:51 10.10.11.1 admin tty 192.168.1.20 stop task_id=1525899775433
timezone=UTC stop_time=1525913570.851 service=shell priv_lvl=15 cmd="configure"
result=success
Mon May 9 17:52:53 10.10.11.1 admin tty 192.168.1.20 stop task_id=1525899775434
timezone=UTC stop_time=1525913573.427 service=shell priv_lvl=15 cmd="interface
1/1/3" result=success
Mon May 9 17:52:54 10.10.11.1 admin tty 192.168.1.20 stop task_id=1525899775435
```

```

timezone=UTC stop_time=1525913574.447 service=shell priv_lvl=15 cmd="no shutdown"
result=success
Mon May 9 17:52:58 10.10.11.1 admin tty 192.168.1.20 stop task_id=1525899775436
timezone=UTC stop_time=1525913578.131 service=shell priv_lvl=15 cmd="ip address
10.10.13.1/24" result=success
Mon May 9 17:52:59 10.10.11.1 admin tty 192.168.1.20 stop task_id=1525899775437
timezone=UTC stop_time=1525913579.468 service=shell priv_lvl=15 cmd="exit"
result=success
Mon May 9 17:53:10 10.10.11.1 admin tty 192.168.1.20 stop task_id=1525899775442
timezone=UTC stop_time=1525913590.204 service=shell priv_lvl=15 cmd="exit"
result=success
Mon May 9 17:53:10 10.10.11.1 admin tty 192.168.1.20 stop task_id=1525899775431
timezone=UTC stop_time=1525913590.205 service=shell priv_lvl=15 result=success
Mon May 9 17:53:44 10.10.11.1 UNKNOWN tty 0.0.0.0 stop task_id=1525899775430
timezone=UTC stop_time=1525913624.473 service=system event=sys_acct
reason="System-accounting-OFF" result=success

```



This sample is representative and not from any particular TACACS+ server implementation.

## Sample REST accounting information on a TACACS+ server

```

Oct 30 16:31:56 10.10.10.1 admin tty 127.0.0.1 start task_id=1540942055868
timezone=UTC start_time=1540942316.36 service=https-server priv_lvl=15
cmd="http-method=POST http-uri=/rest/v1/login" result=success

```



This sample is representative and not from any particular TACACS+ server implementation.

## TACACS+ accounting tasks

The TACACS+ accounting-related tasks are as follows:

Task	Command name	Example
Configuring the accounting sequence for the default connection type	aaa accounting all-mgmt	aaa accounting all-mgmt default start-stop group tg1 tg2 tacacs local
Configuring the accounting sequence for the console connection type	aaa accounting all-mgmt console	aaa accounting all-mgmt console start-stop group tg2 tg3 tacacs local

Task	Command name	Example
Configuring the accounting sequence for the ssh connection type	aaa accounting all-mgmt	aaa accounting all-mgmt ssh start-stop group tg2 tacacs local
Removing remote AAA for the default connection type	aaa accounting all-mgmt	no aaa accounting all-mgmt default start-stop
Showing the accounting configuration	show aaa accounting	show aaa accounting

## Example: Configuring the switch for Remote AAA with TACACS+

### Prerequisites

- TACACS+ servers configured in general according to the information in [Remote AAA TACACS+ server configuration requirements](#). The exact settings appropriate to your environment will vary.
- Logged in to the switch with Administrator privilege and in the `config` context.

### Procedure

1. Configure the global TACACS+ passkey (shared secret) as "xjKW74932qX3j\_\$"

```
switch(config)# tacacs-server key plaintext xjKW74932qX3j_$
switch(config)#
```

2. Add these configuration details for two remote TACACS+ servers:
  - Server 1 with IPv4 address 10.0.0.2, on the management interface (belonging to VRF "mgmt"), using the default PAP protocol.
  - Server 2 with IPv4 address 4.0.0.2, on the data interface (belonging to VRF "default"), using the CHAP protocol.

```
switch(config)# tacacs-server host 10.0.0.2 vrf mgmt
switch(config)# tacacs-server host 4.0.0.2 auth-type chap
switch(config)#
```

3. Create a TACACS+ group named `tac_grp1`, assign TACACS+ server 10.0.0.2 to the group, show the group information.



The default TACACS+ group named `tacacs` includes every TACACS+ server regardless of whether any TACACS+ servers are also assigned to a user-defined TACACS+ group.

```
switch(config)# aaa group server tacacs tac_grp1
switch(config-sg)# server 10.0.0.2 vrf mgmt
switch(config-sg)# exit
switch(config)#
switch(config)# do show aaa server-groups tacacs

***** AAA Mechanism TACACS+ *****
-----
GROUP NAME          | SERVER NAME          | PORT | VRF    | PRIORITY
-----
tac_grp1            | 10.0.0.2             | 49   | mgmt   | 1
-----
tacacs (default)    | 10.0.0.2             | 49   | mgmt   | 1
tacacs (default)    | 4.0.0.2              | 49   | default | 2
-----
switch(config)#
```

4. Define the authentication sequence list so that the new TACACS+ group is first, the default TACACS+ group is second, and local is third. Show the authentication sequence.

```
switch(config)# aaa authentication login default group tac_grp1 tacacs local
switch(config)#
switch(config)# do show aaa authentication
AAA Authentication:
  Fail-through           : Disabled
  Limit Login Attempts   : Not set
  Lockout Time           : 300
  Minimum Password Length : Not set

Default Authentication for All Channels:
-----
---
GROUP NAME          | GROUP PRIORITY
-----
tac_grp1            | 0
tacacs              | 1
local               | 2
-----
---
switch(config)#
```

5. Define the authorization sequence list with two TACACS+ server groups plus local RBAC. Show the authorization sequence.

```
switch(config)# aaa authorization commands default group tac_grp1 tacacs local
switch(config)#
switch(config)# do show aaa authorization

Default command Authorization for All Channels:
-----
```

```

---
GROUP NAME                                | GROUP PRIORITY
-----
--
tac_grp1                                | 0
tacacs                                  | 1
local                                  | 2
-----
--
switch(config)#

```

6. Define the accounting sequence list with two TACACS+ server groups. Show the accounting sequence.

```

switch(config)# aaa accounting all default start-stop group tac_grp1 tacacs
switch(config)#
switch(config)# do show aaa accounting
AAA Accounting:
  Accounting Type                : all
  Accounting Mode                 : start-stop

Default Accounting for All Channels:
-----
--
GROUP NAME                                | GROUP PRIORITY
-----
--
tac_grp1                                | 0
tacacs                                  | 1
-----
--

```

Remote AAA provides the following for your Aruba switch:

- Authentication using remote RADIUS AAA servers. For added security, two-factor authentication may be used. In two-factor authentication, X.509 certificate-based authentication is combined with RADIUS authentication.
- Command authorization is not supported by RADIUS servers, however, user-defined local user groups can be configured with command-authorization rules, providing locally configured per-command authorization for members of such groups. See [User-defined user groups](#).

In the switch default state (without user-defined local groups), basic role-based authorization is available with the three built-in roles (`administrators`, `operators`, `auditors`).

- Transmission of locally collected accounting information to remote RADIUS servers.



For switches that support multiple management modules, all AAA functionality discussed only applies to the active management module. See also *AAA on switches with multiple management modules* in the *High Availability Guide*.

## Default server groups

The switch always has these four default groups:

- `tacacs`: for remote AAA, always contains every configured TACACS+ server.
- `radius`: for remote AAA, always contains every configured RADIUS server.
- `local`: for local authentication.
- `none`: for local (RBAC) authorization.

User-defined AAA servers are always added to the matching default group, either `tacacs` or `radius`. Optionally, each server can also be added to exactly one additional user-defined (custom) group. A maximum of 28 user-defined groups can be created.

The order in which servers are added to a group is important. The server added first is accessed first, and if necessary, the second server is accessed second, and so on.

## Remote AAA (RADIUS) defaults and limits

Setting	Default value / limit
Maximum number of RADIUS servers in an AAA group	16
Maximum number of RADIUS servers that can be configured	16

Setting	Default value / limit
Maximum number of user-defined AAA server groups that can be configured	28
RADIUS authentication	Disabled
RADIUS authentication global timeout	5 seconds
RADIUS authentication passkey (shared secret)	None
RADIUS authentication udp-port	1812
RADIUS global authentication protocol	PAP
RADIUS global retries	1 retry
RADIUS server tracking default interval	300 seconds
RADIUS server access through the default VRF	default*

\*The default value is `default`, unless another VRF is specified during the server configuration.

## About global versus per-RADIUS server passkeys (shared secrets)

To communicate with a RADIUS AAA server, the switch must have a passkey (shared secret) configured that matches what is configured on the server. Use one of these commands to achieve the desired configuration:

- For a global passkey common to every RADIUS server, use `radius-server key`.
- For a per-RADIUS server passkey, use `radius-server host` with the `key` parameter.




---

If both passkeys are configured on the switch, the per-RADIUS server passkey is used.

---

## Remote AAA RADIUS server configuration requirements

The user-supplied RADIUS server must:

- Have an IPv4/IPv6 address or fully qualified domain name (FQDN) that is visible to the switch.
- Have a passkey (shared secret) that matches what is configured on the switch.
- Provide username and password definitions for every switch user. Remote users do not require definition on the switch.
- Configure user role assignment using RADIUS attributes.




---

Consult your RADIUS server documentation for installation and general configuration details.

---





If SSH public key authentication is used, the key information is stored locally on the switch, making username and password definition on the RADIUS server unnecessary.

## User role assignment using RADIUS attributes

User role assignment is configured on the RADIUS server using VSAs (vendor-specific attributes).

RADIUS servers can return multiple attribute value pairs (AVPs) in response to an authentication request.

The attributes are processed in this order of precedence to determine the user role assigned:

- If the `Aruba-Admin-Role` VSA is present, map the user to the matching local user-group name.
  - Else if the `Aruba-Priv-Admin-User` VSA is present, extract the privilege level (1, 15, or 19) and map the user to the local user-group corresponding to this privilege level (1=operators, 15=administrators, 19=auditors). Privilege levels 2 to 14 may also be used with matching local user groups named 2 to 14.
  - Else If Service-Type AVP is present, map `Administrative-User (6)` to administrators and map `NAS-Prompt-User (7)` to operators.
  - Otherwise, the user role cannot be determined, and the authentication fails.

Aruba-Admin-Role	Aruba-Priv-Admin-User	service-type	User role assigned
<GROUP-NAME>	Do not care	Do not care	Matching local user <GROUP-NAME>
Not present	privilege level =1	Do not care	Operators
Not present	privilege level =15	Do not care	Administrators
Not present	privilege level =19	Do not care	Auditors
Not present	privilege level =2 to 14	Do not care	Matching local user groups named 2 to 14
Not present	Not present	Administrative- User (6)	Administrators
Not present	Not present	NAS-Prompt-User (7)	Operators
Not present	Not present	Not present (or = any other value)	None (not authenticated)



The `Service-Type` attribute is retained only for backward compatibility. It is recommended that you instead use the `Aruba-Admin-Role` or `Aruba-Priv-Admin-User` VSA.

## RADIUS server redundancy and access sequence

To prevent authentication interruption, it is common practice to configure more than one RADIUS server.

When identifying RADIUS servers to the switch, server group order (and server order within the group), determines server access order.



When defining the server access sequence for authentication with `aaa authentication login default`, there is an implied `local` included as the last item in the list. If no RADIUS server can be reached, local authentication will be attempted.

## Single source IP address for consistent source identification to AAA servers



If applicable to your installation, it is recommended that you perform the optional configuration mentioned in this section.

If your topology allows the AAA server to be reached through multiple paths, the server interprets the incoming packets to be from different switches even though they are all coming from the same switch. Having a switch associated with multiple IP addresses makes it more difficult to interpret system logs and accounting data.

To ensure that all traffic sent from the switch to the AAA server uses the same source IP address, use `ip source-interface` or `ipv6 source-interface`. These two commands plus the related commands `show ip source-interface` and `show ipv6 source-interface` are described under *Layer 2/3 Interface commands* in the *Command-Line Interface Guide*.

## RADIUS general tasks

General RADIUS tasks, not specific to authentication, are as follows:

Task	Command name	Example
Configuring a RADIUS server	<code>radius-server host</code>	<code>radius-server host 1.1.1.1 vrf default</code> <code>no radius-server host 1.1.1.1 vrf default</code>
Showing global and RADIUS server configurations	<code>show radius-server</code>	<code>show radius-server detail</code>
Configuring a RADIUS server group	<code>aaa group server</code>	<code>aaa group server radius sg3</code> <code>no aaa group server radius sg3</code>
Showing server groups	<code>show aaa server-groups</code>	<code>show aaa server-groups</code>
Adding a RADIUS server to a server-group	<code>server</code>	<code>aaa group server radius sg3</code> <code>server 1.1.1.4 port 32 vrf default</code>
Deleting a RADIUS server from a server-group	<code>server</code>	<code>aaa group server tacacs sg3</code> <code>no server 1.1.1.4 port 32 vrf default</code>
Configuring a RADIUS global passkey	<code>radius-server key</code>	<code>radius-server key plaintext mypasskey123</code>
Configuring PAP or CHAP for RADIUS	<code>radius-server auth-type</code>	<code>radius-server auth-type chap</code> <code>no radius-server auth-type</code>

Task	Command name	Example
Configuring the RADIUS global timeout	<code>radius-server timeout</code>	<code>radius-server timeout 15</code> <code>no radius-server timeout</code>
Configuring the RADIUS global retries	<code>radius-server retries</code>	<code>radius-server retries 3</code> <code>no radius-server retries</code>
Overriding the global retries for a RADIUS server	<code>radius-server host</code>	<code>radius-server host 1.1.1.1 retries 2</code>

## RADIUS authentication

RADIUS authentication occurs as follows:

- User credentials are sent from the switch to RADIUS server using the PAP or CHAP authentication protocol.
- If a user is authenticated, their role is communicated to the switch as Administrator, Operator, or Auditor.
- An unknown user or a user who entered an invalid password is identified as such to the switch, which then rejects user login.

## About authentication fail-through

Normally, authentication is performed by the first AAA server reached. A rarely needed feature named "Authentication fail-through" is available. If Authentication fail-through is enabled and authentication fails on the first reachable AAA server, authentication is attempted on the second AAA server, and so on, until successful authentication or the server list is exhausted.

Enabling Authentication fail-through is typically unnecessary because the user credential databases should be consistent across all AAA servers. Authentication fail-through might be helpful if your AAA user credential databases are not quickly synchronized across all AAA servers.

## RADIUS authentication tasks

The RADIUS authentication-related tasks are as follows:

Task	Command name	Example
Configuring the authentication sequence for the default connection type	<code>aaa authentication login</code>	<code>aaa authentication login default group rg1 rg2 radius local</code>

Task	Command name	Example
Configuring the authentication sequence for the https-server connection type	<code>aaa authentication login</code>	<code>aaa authentication login https-server group rg1 radius local</code>
Removing remote AAA for the default connection type	<code>aaa authentication login</code>	<code>no aaa authentication login default</code>
Configuring authentication fail-through	<code>aaa authentication allow-fail-through</code>	<code>aaa authentication allow-fail-through</code> <code>no aaa authentication allow-fail-through</code>
Showing the authentication sequence	<code>show aaa authentication</code>	<code>show aaa authentication</code>

## Configuring two-factor authentication

Two-factor authentication is available for added security. In two-factor authentication, X.509 certificate-based authentication is combined with RADIUS authentication. When a user establishes an SSH connection to the switch, two factor-authentication occurs as follows:

- The username in the user's X.509 certificate is validated against the local user accounts on the switch.
- The username and password are validated against the accounts on the RADIUS server and the configured trust anchors.

### Prerequisites

- The switch SSH server is enabled.
- Your switch management computer, though its SSH client, is connected to the switch.
- A remote RADIUS server is available to authenticate switch users and is configured on the switch.
- Every user that will use two-factor authentication is configured both on the RADIUS server and locally on the switch using identical usernames. Users are added locally on the switch with the `user` command. These usernames must precisely match the usernames identified by the X.509 user certificates.
- The X.509 CA certificate is both installed on your switch management computer and is also visible to your computer's SSH client. The X.509 CA certificate is the root of trust for the client certificate being used.
- One X.509 certificate per user is available on your switch management computer and is visible to your computer's SSH client. The usernames identified by these user certificates must be the same as the usernames already defined on the RADIUS server and locally on the switch.

### Procedure

1. Create a TA profile with the command `crypto pki ta-profile`. This command switches to the TA configuration context. The TA profile is where the switch stores the root certificate of the CA that is used to validate the certificates of clients communicating with the SSH server.
2. Although optional, it is recommended that you enable certificate revocation checking with the command `revocation-check ocsdp`.
3. Import the root certificate of the CA with the command `ta-certificate`.
4. Exit the TA configuration context with the command `exit`.
5. For each user that will be using two-factor authentication, import the public key from the individual X.509 user certificate with the command `user <USERNAME> authorized-key <PUBKEY>`. Each user identified by <USERNAME> must exist locally on the switch and on the RADIUS authentication server.
6. Enable two-factor authentication with the command `ssh two-factor-authentication`.

## Example

This example installs the root certificate **root-cert** and enables two-factor authentication for user **admin**:

```
switch(config)# crypto pki ta-profile root-cert
switch(config-ta-root-cert)# revocation-check ocsdp
switch(config-ta-root-cert)# ta-certificate
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-ta-cert)# -----BEGIN CERTIFICATE-----
switch(config-ta-cert)# MIIDuTCCAqECCQCuoxeJ2ZNYcjANBgkqhkiG9w0BAQsFADCBq
switch(config-ta-cert)# VVMxEzARBgNVBAgMCkNhbGlmb3JuaWExEDAOBgNVBACMB1JvY
switch(config-ta-cert)# BAoMA0hQTjEVMBMGA1UECwwMSFBOUm9zZXZpbGx1MSowKAYDV
...
switch(config-ta-cert)# x3WFf3dFZ8o9sd5LVAHneH/ztb9MP34z+le1V346r12L2MDL8
switch(config-ta-cert)# BIzD/ST/HaWI+OS+S80rm93PSScEbb9GWk7vshh5E8DH73nW/
switch(config-ta-cert)# 3LvMLZcSSSe5J2Ca2XIhfDme8UaNZ7syGYoCD/TMsAWOnG7yY
switch(config-ta-cert)# -----END CERTIFICATE-----
switch(config-ta-cert)#
The certificate you are importing has the following attributes:
Issuer: C=US, ST=CA, L=Rocklin, O=Company, OU=Site,
       CN=site.com/emailAddress=test.ca@site.com
Subject: C=US, ST=CA, L=Rocklin, O=Company, OU=Site,
        CN=8400/emailAddress=test.ca@site.com
Serial Number: 12121221634631568498 (0xae51217d5945772)

Do you want to accept this certificate (y/n)? y
TA certificate accepted.
switch(config-ta-root-cert)# exit
switch(config)#
switch(config)# user admin authorized-key ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAQAC6krLTrFTnzg3YjLiZKTZEYnh4cUiuOK+cjduxFnZUa
...
iAfcGvqvWtWWBSOwd011DeEZNKn008uEKEtEcAjfrnRHeOk2QJmw== "svl@site.net"
switch(config)#
switch(config)# ssh two-factor-authentication
```

## Secure RADIUS (RadSec)

RADIUS protocol uses UDP as underlying transport layer protocol. RadSec is a protocol that supports RADIUS over TCP and TLS. In conventional RADIUS requests, security is a concern as the confidential data is sent using weak encryption algorithms. The access requests in plain text include information such as user name, IP address and so on. The user password is an encrypted shared secret. As a result, eavesdroppers can listen to these RADIUS requests and collect confidential information. Data protection is

necessary in roaming environments where the RADIUS packets travel across multiple administrative domains and untrusted networks.

RadSec module secures the communication between the switch and RADIUS server using TLS connection. Using RADIUS over TLS provides users with the flexibility to host RADIUS servers across geographies and WAN networks.

For enabling RADIUS security, a CLI option `tls` is provided with the command `radius-server host`, where `tls` stands for Transport Layer Security.

Advantages:

- Secures the communication between the switch and RADIUS server using a TLS session.
- Provides flexibility and enhances security to host RADIUS servers across geographies and WAN networks.
- Uses digital certificates to authenticate both client and server connection.

## RadSec configuration

To configure RadSec protocol, use the following commands:

- Configure TLS using the command `radius-server host tls`.
- Associate the leaf certificate with RadSec feature (`radsec-client`) using the command `crypto pki application`. To use switch inbuilt IDEVID certificate, add `device-identity` with the command `crypto pki application`. By default, switch uses the local certificate for Radsec application. For more information on installing certificates, see [PKI](#) chapter.



---

RadSec mandates validating server certificates SAN/CN while establishing connections.

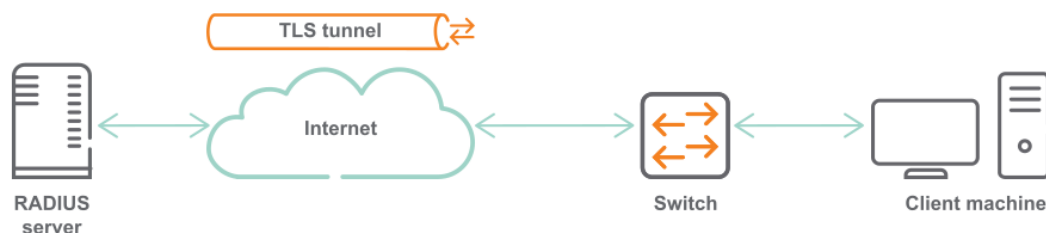
---

## Deployment scenarios

You can deploy the RADIUS/TLS servers in any of the following scenarios:

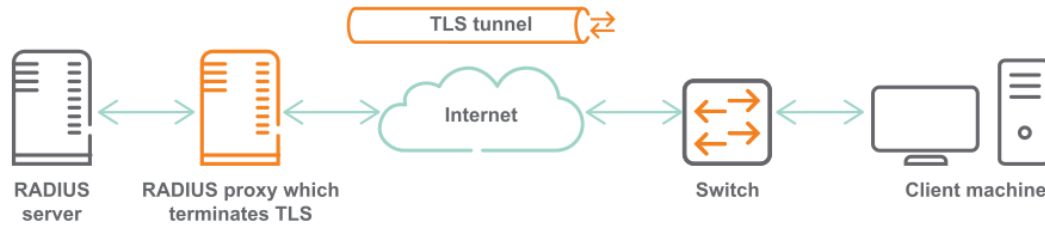
- Scenario 1: Switch establishes TLS connection with the RADIUS server.
- Scenario 2: Switch establishes TLS connection with the proxy server, which communicates with the RADIUS server.

**Figure 1** Scenario 1: Switch establishes TLS connection with the RADIUS server



In this scenario, the RADIUS server is across WAN. The RADIUS/TLS secures the user data by creating an encrypted TLS tunnel between the switch and authentication server.

**Figure 2** Scenario 2: Switch establishes TLS connection with the proxy server, which communicates with the RADIUS server



In this scenario, multiple RADIUS servers are distributed over WAN (untrusted networks). RADIUS proxy directs the RADIUS requests to the RADIUS server, which listens on UDP. The proxy server uses the switch certificates to authenticate the client-server credentials. As a result, all RADIUS communications across the network are TLS encrypted.

## Example of RadSec configuration

### Prerequisite

- ClearPass version is 6.7.4 or higher.

### ClearPass as RadSec server

Following are the steps to configure ClearPass as RadSec server:

1. From the ClearPass Web UI, navigate to **Administration > Certificates > Certificate store** and click **Import Certificate** to import the Root CA certificate to the ClearPass certificate store.

2. The **Import Certificate** window opens. In the **Certificate Type** field, select **Server Certificate**. Specify the server and upload method. In the **Usage** field, you must select **Radsec Server**

## Certificate.



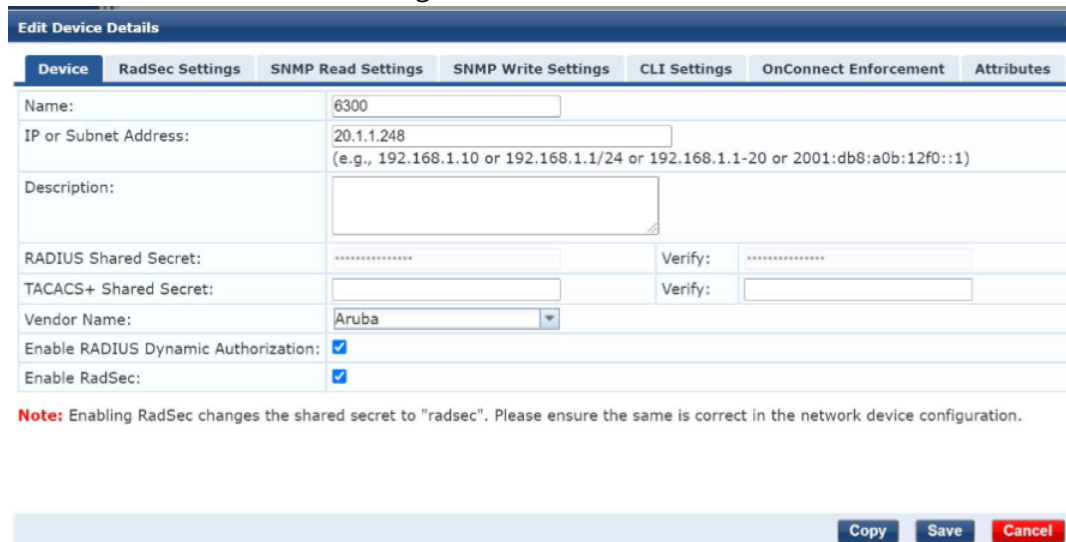
The **Import Certificate** dialog box contains the following fields and controls:

- Certificate Type:** Server Certificate (dropdown)
- Server:** CPPM2 (dropdown)
- Type:** RadSec Server Certificate (dropdown)
- Upload Method:** Upload Certificate and Use Saved Private Key (dropdown)
- Certificate File:** [Text field] **Browse...** (button)

**Note:** Certificates with a wildcard as the common name (ex: \*.arubanetworks.com) and Extended Validation certificates (EV, "Green Bar") are not recommended for use as the RADIUS/EAP server certificate. Some clients may be unable to authenticate when these types of certificates are used.

**Import** (button) **Cancel** (button)

3. Click **Import**.
4. next, click **Create Certificate Signing Request**. In the **Common Name** field, enter the IP address of the ClearPass server. For configuring `radius-server host`, enter the hostname.
5. Click **Submit**. You can download the CSR or copy and paste the displayed CSR content into the web form in the enrollment process.
6. Sign the created CSR with the CA.
7. Select **Enable RadSec** while adding devices. The IP address is used as the source IP of the switch.



The **Edit Device Details** dialog box has tabs for **Device**, **RadSec Settings**, **SNMP Read Settings**, **SNMP Write Settings**, **CLI Settings**, **OnConnect Enforcement**, and **Attributes**. The **Device** tab is active, showing the following fields:

- Name:** 6300
- IP or Subnet Address:** 20.1.1.248 (e.g., 192.168.1.10 or 192.168.1.1/24 or 192.168.1.1-20 or 2001:db8:a0b:12f0::1)
- Description:** [Text field]
- RADIUS Shared Secret:** [Text field] **Verify:** [Text field]
- TACACS+ Shared Secret:** [Text field] **Verify:** [Text field]
- Vendor Name:** Aruba (dropdown)
- Enable RADIUS Dynamic Authorization:** ☒
- Enable RadSec:** ☒

**Note:** Enabling RadSec changes the shared secret to "radsec". Please ensure the same is correct in the network device configuration.

**Copy** (button) **Save** (button) **Cancel** (button)

## RADIUS accounting

This accounting information is captured and made available for sending to remote accounting servers:

- Port access accounting
- Exec Accounting: user login/logout events
- Command accounting: commands executed by users. The Vendor-Specific Attribute (VSA) `Aruba_Command_String` with a value of 46 is available.
- System accounting: remote accounting On/Off events.
- CLI show commands.
- Interactions on the non-CLI interfaces: REST and WebUI.





---

With RADIUS, command accounting logs a maximum of 247 characters per command entered by the user.

---

The following is not captured or made available as accounting information:

- CLI commands that reboot the switch.
- Interactions in the bash shell.



---

Local accounting (always enabled) must be functioning properly for remote Accounting to work.

---



---

The accounting information is sent to the first reachable remote RADIUS AAA server (configured for remote accounting). If no remote RADIUS server is reachable, local accounting remains available.

---

## Sample general accounting information

```
~~~~~ EXEC ~~~~~~

Mon Jul 16 16:25:27 2018
  User-Name = "admin"
  NAS-Identifier = "switchx"
  NAS-Port = 331
  NAS-Port-Type = Virtual
  Acct-Status-Type = Start
  Acct-Session-Id = "1531769192494"
  Acct-Authentic = Local
  Calling-Station-Id = "0.0.0.0"
  Event-Timestamp = "Jul 16 2018 16:25:22 PDT"
  Acct-Delay-Time = 0
  NAS-IP-Address = 10.10.10.1
  Acct-Unique-Session-Id = "b83e29f4140c17b1"
  Timestamp = 1531783527

~~~ EXEC stop ~~~

Mon Jul 16 16:26:42 2018
  User-Name = "admin"
  NAS-Identifier = "switchx"
  NAS-Port = 331
  NAS-Port-Type = Virtual
  Acct-Status-Type = Stop
  Acct-Session-Id = "1531769192494"
  Acct-Authentic = Local
  Calling-Station-Id = "0.0.0.0"
  Event-Timestamp = "Jul 16 2018 16:26:37 PDT"
  Acct-Delay-Time = 0
  Acct-Session-Time = 75
  NAS-IP-Address = 10.10.10.1
  Acct-Unique-Session-Id = "b83e29f4140c17b1"
  Timestamp = 1531783602

~~~~~ CMD ACCOUNTING ~~~~~~

Mon Jul 16 16:26:42 2018
  User-Name = "admin"
  NAS-Identifier = "switchx"
  NAS-Port = 331
  NAS-Port-Type = Virtual
  Acct-Status-Type = Stop
  Acct-Session-Id = "1531769192496"
```

```

Acct-Authentic = Local
Aruba-Command-String = "exit"
Calling-Station-Id = "0.0.0.0"
Event-Timestamp = "Jul 16 2018 16:26:37 PDT"
Acct-Delay-Time = 0
NAS-IP-Address = 10.10.10.1
Acct-Unique-Session-Id = "280710992629128c"
Timestamp = 1531783602

~~~~~ SYSTEM ACCOUNTING ~~~~~

Mon Jul 16 17:13:02 2018
  User-Name = "UNKNOWN"
  NAS-Identifier = "UNKNOWN"
  NAS-Port = 331
  NAS-Port-Type = Virtual
  Acct-Status-Type = Accounting-On
  Acct-Session-Id = "1531769192506"
  Acct-Authentic = Local
  Calling-Station-Id = "0.0.0.0"
  Event-Timestamp = "Jul 16 2018 17:12:56 PDT"
  Acct-Delay-Time = 0
  NAS-IP-Address = 10.10.10.1
  Acct-Unique-Session-Id = "b478e6402c86933e"
  Timestamp = 1531786382

Mon Jul 16 17:12:55 2018
  User-Name = "UNKNOWN"
  NAS-Identifier = "UNKNOWN"
  NAS-Port = 331
  NAS-Port-Type = Virtual
  Acct-Status-Type = Accounting-Off
  Acct-Session-Id = "1531769192491"
  Acct-Authentic = Local
  Calling-Station-Id = "0.0.0.0"
  Event-Timestamp = "Jul 16 2018 17:12:49 PDT"
  Acct-Delay-Time = 0
  NAS-IP-Address = 10.10.10.1
  Acct-Unique-Session-Id = "93da1f094121f2ee"
  Timestamp = 1531786375

~~~~~

```




---

This sample is representative and not from any particular RADIUS server implementation.

---

## RADIUS accounting tasks

The RADIUS accounting-related tasks are as follows:

Task	Command name	Example
Configuring the accounting sequence for the default connection type	aaa accounting all-mgmt	aaa accounting all-mgmt default start-stop group rg1 rg2 radius local
Configuring the accounting sequence for the https-server connection type	aaa accounting all-mgmt	aaa accounting all-mgmt https-server start-stop group rg1 radius local
Removing remote AAA for the default connection type	aaa accounting all-mgmt	no aaa accounting all-mgmt default start-stop
Showing the accounting configuration	show aaa accounting	show aaa accounting

## Example: Configuring the switch for Remote AAA with RADIUS

### Prerequisites

- RADIUS servers configured in general according to the information in [Remote AAA RADIUS server configuration requirements](#). The exact settings appropriate to your environment will vary.
- Logged in to the switch with Administrator privilege and in the `config` context.

### Procedure

1. Configure the global RADIUS passkey (shared secret) as "xjKW74932qX3j\_\$"

```
switch(config)# radius-server key plaintext xjKW74932qX3j_$
switch(config)#
```

2. Add these configuration details for two remote RADIUS servers.
  - Server 1 with IPv4 address 10.0.0.2, on the management interface (belonging to VRF "mgmt"), using the default PAP protocol.
  - Server 2 with IPv4 address 4.0.0.2, on the data interface (belonging to VRF "default"), using the CHAP protocol.

```
switch(config)# radius-server host 10.0.0.2 vrf mgmt
switch(config)# radius-server host 4.0.0.2 auth-type chap
switch(config)#
```

3. Create a RADIUS group named `rad_grp1`, assign RADIUS server 10.0.0.2 to the group, show the group information.



The default RADIUS group named `radius` includes every RADIUS server regardless of whether any RADIUS servers are also assigned to a user-defined RADIUS group.

```
switch(config)# aaa group server radius rad_grp1
switch(config-sg)# server 10.0.0.2 vrf mgmt
switch(config-sg)# exit
switch(config)#
switch(config)# do show aaa server-groups radius

***** AAA Mechanism RADIUS *****

-----
GROUP NAME          | SERVER NAME          | PORT | VRF    | PRIORITY
-----
rad_grp1            | 10.0.0.2             | 1812 | mgmt   | 1
-----
radius (default)    | 10.0.0.2             | 1812 | mgmt   | 1
radius (default)    | 4.0.0.2              | 1812 | default| 2
-----
switch(config)#
```

4. Define the authentication sequence list so that the new RADIUS group is first, the default RADIUS group is second, and local is third. Show the authentication sequence.

```
switch(config)# aaa authentication login default group rad_grp1 radius local
switch(config)#
switch(config)# do show aaa authentication

AAA Authentication:
  Fail-through           : Disabled
  Limit Login Attempts   : Not set
  Lockout Time           : 300
  Minimum Password Length : Not set

Default Authentication for All Channels:
-----
-
GROUP NAME          | GROUP PRIORITY
-----
-
rad_grp1            | 0
radius              | 1
local               | 2
-----
-
switch(config)#
```

5. Define the accounting sequence list with two RADIUS server groups. Show the accounting sequence.

```
switch(config)# aaa accounting all default start-stop group rad_grp1 radius
switch(config)#
switch(config)# do show aaa accounting
AAA Accounting:
  Accounting Type           : all
  Accounting Mode           : start-stop
```

Default Accounting for All Channels:

GROUP NAME	GROUP PRIORITY
rad_grp1	0
radius	1

### aaa accounting all-mgmt

#### Syntax

```
aaa accounting all-mgmt <CONNECTION-TYPE> start-stop {local | group <GROUP-LIST>}  
no aaa accounting all-mgmt <CONNECTION-TYPE>
```

#### Description

Defines accounting as being local (with the name `local`) (the default). Or defines a sequence of remote AAA server groups to be accessed for accounting purposes.

For remote accounting, the information is sent to the first reachable remote server that was configured with this command for remote accounting. If no remote server is reachable, local accounting remains available. Each available connection type (channel) can be configured individually as either local or using remote AAA server groups. All server groups named in your command, must exist. This command can be issued multiple times, once for each connection type. Local is always available for any connection type not configured for remote accounting.



---

The system accounting log is not associated with any connection type (channel) and is therefore sent to the accounting method configured on the default connection type (channel) only.

---

The `no` form of this command removes for the specified connection type, any defined remote AAA server group accounting sequence. Local accounting is available for connection types without a configured remote AAA server group list (whether default or for the specific connection type).

#### Command context

`config`

#### Parameters

`<CONNECTION-TYPE>`

One of these connection types (channels):

`default`

Defines a list of accounting server groups to be used for the `default` connection type. This configuration applies to all other connection types (`console`, `https-server`, `ssh`) that are not explicitly configured with this command. For example, if you do not use `aaa accounting all-mgmt console...` to define the console accounting list, then this default configuration is used for console.

`console`

Defines a list of accounting server groups to be used for the `console` connection type.

`https-server`

Defines a list of accounting server groups to be used for the `https-server` (REST, Web UI) connection type.

`ssh`

Defines a list of accounting server groups to be used for the `ssh` connection type.

`start-stop`

Selects accounting information capture at both the beginning and end of a process.

`local`

Selects local-only accounting when used without the `group` parameter.

`group <GROUP-LIST>`

Specifies the list of remote AAA server group names. Each name can be specified one time. Predefined remote AAA group names `tacacs` and `radius` are available. Although not a group name, predefined name `local` is available. User-defined TACACS+ and RADIUS server group names may also be used. The remote AAA server groups are accessed in the order that the group names are listed in this command. Within each group, the servers are accessed in the order in which the servers were added to the group. Server groups are defined using command `aaa group server` and servers are added to a server group with the command `server`.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

Local accounting is always active. It cannot be turned off.

## Examples

Defining the default accounting sequence based on two user-defined TACACS+ server groups, then the default TACACS+ server group, and finally (if needed), local accounting.

```
switch(config)# aaa accounting all-mgmt default start-stop group tg1 tg2 tacacs  
local
```

Defining the console accounting sequence based on two user-defined TACACS+ server groups, then the default TACACS+ server group, and finally (if needed), local accounting.

```
switch(config)# aaa accounting all-mgmt console start-stop group tg2 tg3 tacacs  
local
```

Defining the ssh accounting sequence based on one user-defined TACACS+ server group and then the default TACACS+ server group.

```
switch(config)# aaa accounting all-mgmt ssh start-stop group tg2 tacacs
```

Defining the default accounting sequence based on two user-defined RADIUS server groups, then the default RADIUS server group, and finally (if needed), local accounting.

```
switch(config)# aaa accounting all-mgmt default start-stop group rg1 rg2 radius  
local
```

Defining the https-server accounting sequence based on one user-defined RADIUS server group and then the default RADIUS server group.

```
switch(config)# aaa accounting all-mgmt https-server start-stop group rg1 radius
```

Setting local accounting for the default connection type:

```
switch(config)# aaa accounting all-mgmt default start-stop local
```

## aaa authentication allow-fail-through

### Syntax

```
aaa authentication allow-fail-through  
no aaa authentication allow-fail-through
```

### Description

Enables authentication fail-through. When this option is enabled, the next server/authentication method is tried after an authentication failure.

The `no` form of this command disables authentication fail-through. If the system fails to authenticate with a reachable TACACS+ or RADIUS server, the system does not attempt to authenticate with the next TACACS+/RADIUS server.

### Command context

config

### Authority

Administrators or local user group members with execution rights for this command.

### Example

Enabling authentication fail-through:

```
switch(config)# aaa authentication allow-fail-through
```

## aaa authentication login

### Syntax

```
aaa authentication login <CONNECTION-TYPE> {local | group <GROUP-LIST>}  
no aaa authentication login <CONNECTION-TYPE>
```

### Description

Defines authentication as being local (with the name `local`) (the default). Or defines a sequence of remote AAA server groups to be accessed for authentication purposes. Each available connection type (channel) can be configured individually as either local or using remote AAA server groups. All server groups named in your command, must exist. This command can be issued multiple times, once for each connection type. Local is always available for any connection type not configured for remote AAA authentication.



---

If you do not want local authentication to occur in cases where all AAA servers contacted reject the user's credentials, do not enable authentication fail-through (command `aaa authentication allow-fail-through`).

---

The `no` form of this command removes for the specified connection type, any defined remote AAA server group authentication sequence. Local authentication is available for connection types without a configured remote AAA server group list (whether default or for the specific connection type).

### Command context



config

## Parameters

<CONNECTION-TYPE>

One of these connection types (channels):

`default`

Defines a list of authentication server groups to be used for the `default` connection type. This configuration applies to all other connection types (`console`, `https-server`, `ssh`) that are not explicitly configured with this command. For example, if you do not use `aaa authentication login console...` to define the console authentication list, then this default configuration is used for `console`.

`console`

Defines a list of authentication server groups to be used for the `console` connection type.

`https-server`

Defines a list of authentication server groups to be used for the `https-server` (REST, Web UI) connection type.

`ssh`

Defines a list of authentication server groups to be used for the `ssh` connection type.

`local`

Selects local-only authentication when used without the `group` parameter.

`group` <GROUP-LIST>

Specifies the list of remote AAA server group names. Each name can be specified one time. Predefined remote AAA group names `tacacs` and `radius` are available. Although not a group name, predefined name `local` is available. User-defined TACACS+ and RADIUS server group names may also be used. The remote AAA server groups are accessed in the order that the group names are listed in this command. Within each group, the servers are accessed in the order in which the servers were added to the group. Server groups are defined using command `aaa group server` and servers are added to a server group with the command `server`.

If no AAA server is reachable, local authentication is attempted.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Defining the default authentication sequence based on two user-defined TACACS+ server groups, then the default TACACS+ server group, and finally (if needed), local authentication.

```
switch(config)# aaa authentication login default group tg1 tg2 tacacs local
```

Defining the console authentication sequence based on two user-defined TACACS+ server groups, then the default TACACS+ server group, and finally (if needed), local authentication.

```
switch(config)# aaa authentication login console group tg2 tg3 tacacs local
```

Defining the ssh authentication sequence based on one user-defined TACACS+ server group and then the default TACACS+ server group.

```
switch(config)# aaa authentication login ssh group tg2 tacacs
```

Defining the default authentication sequence based on two user-defined RADIUS server groups, then the default RADIUS server group, and finally (if needed), local authentication.

```
switch(config)# aaa authentication login default group rg1 rg2 radius local
```

Defining the https-server authentication sequence based on one user-defined RADIUS server group and then the default RADIUS server group.

```
switch(config)# aaa authentication login https-server group rg1 radius
```

Setting local authentication for the default connection type:

```
switch(config)# aaa authentication login default local
```

## aaa authorization commands

### Syntax

```
aaa authorization commands <CONNECTION-TYPE> {local | none}  
aaa authorization commands <CONNECTION-TYPE> group <GROUP-LIST>
```

```
no aaa authorization commands <CONNECTION-TYPE>
```

### Description

Defines authorization as being basic local RBAC (specified as `none`), or as full-fledged local RBAC specified as `local` (the default), or as remote TACACS+ (specified with `group <GROUP-LIST>`). Each available connection type (channel) can be configured individually. All server groups named in the command, must exist. This command can be issued multiple times, once for each connection type.

The `no` form of this command unconfigures authorization for the specified connection type, reverting to the default of `local`.



---

Although only TACACS+ servers are supported for remote authorization, local authorization (basic or full-fledged) can be used with remote RADIUS authentication.

---

### Command context

config

### Parameters

<CONNECTION-TYPE>

One of these connection types (channels):

`default`

Selects the `default` connection type for configuration. This configuration applies to all other connection types (`console`, `ssh`) that are not explicitly configured with this command. For example, if you do not use `aaa authorization commands console...` to define the console authorization list, then this default configuration is used for console.

`console`

Selects the `console` connection type for configuration.

`ssh`

Selects the `ssh` connection type for configuration.

## local (the default)

When used alone without `group <GROUP-LIST>`, selects local authorization which can be used to provide authorization for a purely local setup without any remote AAA servers and also for when RADIUS is used for remote Authentication and Accounting but Authorization is local.

When used after `group`, provides for fallback (to full-fledged local authorization) when every server in every specified TACACS+ server group cannot be reached.



---

If any TACACS+ server in the specified groups is reachable, but the command fails to be authorized by that server, the command is rejected and local authorization is never attempted. Local authorization is only attempted if every TACACS+ server cannot be reached.

---

## none

When used alone without `group <GROUP-LIST>`, selects basic local RBAC authorization, for use with the built-in user groups (administrators, operators, auditors).

When used after `group`, provides for fallback (to basic local RBAC authorization) when every server in every specified TACACS+ server group cannot be reached.



---

With `none`, for users belonging to user-defined user groups, all commands can be executed regardless of what authorization rules are defined in such groups. For per-command local authorization, use `local` instead.

---

## group <GROUP-LIST>

Specifies the list of remote AAA server group names. Predefined remote AAA group name `tacacs` is available. User-defined TACACS+ server group names may also be used. The remote AAA server groups are accessed in the order that the group names are listed in this command. Within each group, the servers are accessed in the order in which the servers were added to the group. Server groups are defined using command `aaa server group` and servers are added to a server group using command `server`.

It is recommended to always include either the special name `local` or `none` as the last name in the group list. If both `local` and `none` are omitted, and no remote AAA server is reachable (or the first reachable server cannot authorize the command), command execution for the current user will not be possible.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

### TACACS+ server authorization considerations



---

Use caution when configuring authorization, as it has no fail through. If the switch is not configured properly, the switch might get into an unusable state in which all command execution is prohibited.

---

To prevent authorization difficulties:

- Make sure that all listed TACACS+ servers can authorize users for command execution.
- Make sure that credential database changes are promptly synchronized across all TACACS+ servers.
- Make sure either `local` or `none` is included as the last name in the group list. If both `local` and `none` are omitted, and no remote TACACS+ server is reachable (or the first reachable server cannot authorize), authorization will not be possible.
- Although not recommended, if you choose to omit both `local` and `none` from the list, and are manipulating configuration files, special caution is necessary. If the source configuration includes TACACS+ authorization and you are copying configuration from an existing switch into the running

configuration of a new switch, and you have not yet configured the interface or routing information to reach the TACACS+ server, the switch will enter an unusable state, requiring hard reboot.

To avoid getting into this situation that can occur when `local` and `none` have been omitted, do either of the following:

- In the configuration source, delete or comment-out the line configuring remote authorization. Then, after the configuration copy and paste, manually configure authorization.
- Move the line configuring the authorization to the end of the source configuration before copying and pasting.

## Examples

Defining the default authorization sequence based on a user-defined TACACS+ server group, then the default TACACS+ server group, and finally (as a precaution), `local` authorization:

```
switch(config)# aaa authorization commands default group tg1 tacacs local  
All commands will fail if none of the servers in the group list are reachable.  
Continue (y/n)? y
```

Defining the console authorization sequence based on two user-defined TACACS+ server groups, and finally (as a precaution), `local` authorization:

```
switch(config)# aaa authorization commands console group tg1 tg2 local  
All commands will fail if none of the servers in the group list are reachable.  
Continue (y/n)? y
```

Setting the authorization for default to `local`:

```
switch(config)# aaa authorization commands default local
```

Setting the authorization for the SSH interface to `none`:

```
switch(config)# aaa authorization commands ssh none
```

## aaa group server

### Syntax

```
aaa group server {tacacs | radius} <SERVER-GROUP-NAME>
```

```
no aaa group server {tacacs | radius} <SERVER-GROUP-NAME>
```

### Description

Creates an AAA server group that is either empty or contains preconfigured RADIUS/TACACS+ servers. You can create a maximum of 28 server groups.

The `no` form of this command deletes a server group. Only a preconfigured user-defined RADIUS/TACACS+ server group can be deleted. RADIUS or TACACS+ servers that were in a deleted server group remain a part of their default server group. The default server group for TACACS+ servers is `tacacs`. The default server group for RADIUS servers is `radius`.

### Command context

config

## Parameters

server {tacacs | radius}

Select either `tacacs` or `radius` for the server type.

<SERVER-GROUP-NAME>

Specifies the name of the server group to be created. The name of the server group can have a maximum of 32 characters.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Creating TACACS+ server group sg1:

```
switch(config)# aaa group server tacacs sg1
```

Creating RADIUS server group sg3:

```
switch(config)# aaa group server radius sg3
```

Deleting TACACS+ server group sg1:

```
switch(config)# no aaa group server tacacs sg1
```

Deleting RADIUS server group sg3:

```
switch(config)# no aaa group server radius sg3
```

# radius-server auth-type

## Syntax

```
radius-server auth-type {pap | chap}  
no radius-server auth-type
```

## Description

Enables the CHAP or PAP authentication protocol, which is used for communication with the RADIUS servers, at the global level. You can override this command with a fine-grained per server `auth-type` configuration.

The `no` form of this command resets the global authentication mechanism for RADIUS to PAP, which is the default authentication mechanism for RADIUS.

## Command context

config

## Parameters

auth-type {pap | chap}

Selects either the PAP or CHAP authentication protocol.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Authenticating CHAP:

```
switch(config)# radius-server auth-type chap
```

Authenticating PAP:

```
switch(config)# radius-server auth-type pap
```

# radius-server host

## Syntax

```
radius-server host {<FQDN> | <IPv4> | <IPv6>}  
    [key [plaintext <PASSKEY> | ciphertext <PASSKEY>]]  
    [timeout <TIMEOUT-SECONDS>] [port <PORT-NUMBER>]  
    [auth-type {pap | chap}] [acct-port <ACCT-PORT>] [retries <RETRY-COUNT>]  
    [tracking {enable | disable}] [tracking-mode {any | dead-only}] [vrf <VRF-NAME>]  
  
no radius-server host {<FQDN> | <IPv4> | <IPv6>} [port <PORT-NUMBER>] [vrf <VRF-NAME>]
```

## Description

Adds a RADIUS server. By default, the RADIUS server is associated with the server group named `radius`. The `no` form of this command removes a previously added RADIUS server.



---

For enhanced security with IPsec, the alternative command `radius-server host secure ipsec` is available. The standard non-IPsec `radius-server host` command does not modify any existing IPsec configuration. If IPsec is already configured for the RADIUS server, then IPsec will remain enabled for the server.

---

## Command context

config

## Parameters

{<FQDN> | <IPv4> | <IPv6>}

Specifies the RADIUS server as:

- <FQDN>: a fully qualified domain name.
- <IPv4>: an IPv4 address.
- <IPv6>: an IPv6 address.

key [plaintext <PASSKEY> | ciphertext <PASSKEY>]

Specifies either a plaintext or an encrypted local shared-secret passkey for the server. As per RFC 2865, the shared-secret can be a mix of alphanumeric and special characters. Plaintext passkeys are between 1 and 32 alphanumeric and special characters.



---

When `key` is entered without either sub-parameter, plaintext passkey prompting occurs upon pressing Enter. Enter must be pressed immediately after the `key` parameter without entering other parameters. The entered passkey characters are masked with asterisks.

When `key` is omitted, the server uses the global passkey. This command requires either the global or local passkey to be set; otherwise the server will not be contacted. Command `radius-server key` is available for setting the global passkey.

---

`timeout <TIMEOUT-SECONDS>`

Specifies the timeout. The range is 1 to 60 seconds. If a timeout is not specified, the value from the global timeout for RADIUS is used.

`port <PORT-NUMBER>`

Specifies the authentication port number. Range: 1 to 65535. Default RADIUS: 1812.

`auth-type {pap | chap}`

Selects either PAP (default) or CHAP authentication type. If this parameter is not specified, the RADIUS global default is used.

`acct-port <ACCT-PORT>`

Specifies the UDP accounting port number. Range: 1 to 65535. Default: 1813.

`retries <RETRY-COUNT>`

Specifies the number of retry attempts for contacting the specified RADIUS server. Range is 0 to 5 attempts. If no retry value is provided, the default value of 1 is used.

`tracking {enable | disable}`

Enables or disables server tracking for the RADIUS server. Tracked servers are probed at the start of each server tracking interval to check if they are reachable.

Use command `radius-server tracking` to configure RADIUS server tracking globally.



---

Server tracking uses authentication request and response packets to determine server reachability status. The server tracking user name and password are used to form the request packet which is sent to the server with tracking enabled. Upon receiving a response to the request packet, the server is considered to be reachable.

---

`tracking-mode {any | dead-only}`

Configures tracking mode for the RADIUS server that has tracking enabled with the server. The tracking mode is used to monitor the status of RADIUS server reachability. The default tracking mode is `any`.

Sets the tracking mode to:

- `any`: track the RADIUS server irrespective of its server reachability.
- `dead-only`: track the RADIUS server only when the server is marked as unreachable.

`vrf VRF-NAME>`

Specifies the VRF name to be used for communicating with the server. If no VRF name is provided, the default VRF named `default` is used.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

If the fully qualified domain name is provided for the RADIUS server, a DNS server must be configured and accessible through the same VRF which is configured for the RADIUS server. This configuration is required for the resolution of the RADIUS server hostname to its IP address. If a DNS server is not available for this VRF, the RADIUS servers reachable through this VRF must be configured by means of their IP addresses only.

## Examples

Adding a RADIUS server with an IPv4 address and a prompted passkey:

```
switch(config)# radius-server host 1.1.1.5 key
Enter the RADIUS server key: *****
Re-Enter the RADIUS server key: *****
```

Adding a RADIUS server with an IPv4 address and a named VRF:

```
switch(config)# radius-server host 1.1.1.1 vrf mgmt
```

Adding a RADIUS server with an IPv4 address, a port, and a named VRF:

```
switch(config)# radius-server host 1.1.1.2 port 32 vrf mgmt
```

Adding a RADIUS server with an FQDN, a timeout, port number, and a named VRF:

```
switch(config)# radius-server host abc.com timeout 15 port 32 vrf vrf_blue
```

Adding a RADIUS server with an IPv6 address:

```
switch(config)# radius-server host 2001:0db8:85a3:0000:0000:8a2e:0370:7334
```

Adding a RADIUS server with tracking enabled and tracking mode is set to dead-only:

```
switch(config)# radius-server host 1.1.1.1 tracking enable tracking-mode dead-only
```

Adding a RADIUS server with tracking disabled:

```
switch(config)# radius-server host 1.1.1.1 tracking disable
```

Adding a RADIUS server with an IPv4 address, key, encrypted passkey, number of retries, and VRF name:

```
switch(config)# radius-server host 1.1.1.6 key ciphertext AQBapStbgHt1X2JlbcEcQ1
xbbzWjrFr9UsfH3+00x5Qj0qcQBAAAAJ5WZBQ= retries 3 vrf vrf_red
```

Deleting a RADIUS server with an IPv4 address and specified VRF:

```
switch(config)# no radius-server host 1.1.1.1 vrf mgmt
```

Deleting a RADIUS server with an FQDN, port, and specified VRF:

```
switch(config)# no radius-server host abc.com port 32 vrf vrf_blue
```

## radius-server host secure ipsec



## Syntax

Syntax for a RADIUS server that uses IPsec for authentication:

```
radius-server host {<FQDN> | <IPV4> | <IPV6>}
    [key [plaintext <PASSKEY> | ciphertext <PASSKEY>]]
    [timeout <TIMEOUT-SECONDS>] [port <PORT-NUMBER>]
    [auth-type {pap | chap}] [acct-port <ACCT-PORT>] [retries <RETRY-COUNT>]
    [tracking {enable | disable}] [tracking-mode {any | dead-only}] [vrf <VRF-NAME>]
    secure ipsec authentication spi <SPI-INDEX> <AUTH-TYPE> <AUTH-KEY-TYPE> [<AUTH-KEY>]

no radius-server host {<FQDN> | <IPV4> | <IPV6>} [port <PORT-NUMBER>]
    [vrf <VRF-NAME>] secure ipsec authentication
```

Syntax for a RADIUS server that uses IPsec for both authentication and encryption:

```
radius-server host {<FQDN> | <IPV4> | <IPV6>}
    [key [plaintext <PASSKEY> | ciphertext <PASSKEY>]]
    [timeout <TIMEOUT-SECONDS>] [port <PORT-NUMBER>]
    [auth-type {pap | chap}] [acct-port <ACCT-PORT>] [retries <RETRY-COUNT>]
    [tracking {enable | disable}] [tracking-mode {any | dead-only}] [vrf <VRF-NAME>]
    secure ipsec encryption spi <SPI-INDEX> <AUTH-TYPE> <AUTH-KEY-TYPE>
    [<AUTH-KEY>] <ENCRYPT-TYPE> <ENCRYPT-KEY-TYPE> [<ENCRYPT-KEY>]

no radius-server host {<FQDN> | <IPV4> | <IPV6>} [port <PORT-NUMBER>]
    [vrf <VRF-NAME>] secure ipsec encryption
```

## Description

Adds a RADIUS server that uses IPsec for enhanced security (authentication and possibly encryption). By default, the RADIUS server is associated with the server group named `radius`.

The `no` form of this command removes a previously added RADIUS (with IPsec) server.



---

Unless enhanced security with IPsec is required, use the `radius-server host` command instead.

---

## Command context

config

## Parameters

{<FQDN> | <IPV4> | <IPv6>}

Specifies the RADIUS server as:

- <FQDN>: a fully qualified domain name.
- <IPV4>: an IPv4 address.
- <IPV6>: an IPv6 address.

key [plaintext <PASSKEY> | ciphertext <PASSKEY>]

Selects either a plaintext or an encrypted local shared-secret passkey for the server. As per RFC 2865, shared-secret can be a mix of alphanumeric and special characters. Plaintext passkeys are between 1 and 32 alphanumeric and special characters.



---

When `key` is entered without either sub-parameter, plaintext passkey prompting occurs upon pressing Enter. Enter must be pressed immediately after the `key` parameter without entering other parameters. The entered passkey characters are masked with asterisks.

When `key` is omitted, the server uses the global passkey. This command requires either the global or local passkey to be set; otherwise the server will not be contacted. Command `radius-server key` is available for setting the global passkey.

---

`timeout <TIMEOUT-SECONDS>`

Specifies the timeout. The range is 1 to 60 seconds. If a timeout is not specified, the value from the global timeout for RADIUS is used.

`port <PORT-NUMBER>`

Specifies the authentication port number. Range: 1 to 65535. Default: 1812.

`auth-type {pap | chap}`

Selects either the PAP (the default) or CHAP authentication types. If this parameter is not specified, the RADIUS global default is used.

`acct-port <ACCT-PORT>`

Specifies the UDP accounting port number. Range: 1 to 65535. Default: 1813.

`retries <RETRY-COUNT>`

Specifies the number of retry attempts for contacting the specified RADIUS server. Range is 0 to 5 attempts. If no retry value is provided, the default value of 1 is used.

`tracking {enable | disable}`

Enables or disables server tracking for the RADIUS server. Tracked servers are probed at the start of each server tracking interval to check if they are reachable.

Use command `radius-server tracking` to configure RADIUS server tracking globally.



---

Server tracking uses authentication request and response packets to determine server reachability status. The server tracking user name and password are used to form the request packet which is sent to the server with tracking enabled. Upon receiving a response to the request packet, the server is considered to be reachable.

---

`tracking-mode {any | dead-only}`

Configures tracking mode for the RADIUS server that has tracking enabled with the server. The tracking mode is used to monitor the status of RADIUS server reachability. The default tracking mode is `any`.

Sets the tracking mode to:

- `any`: track the RADIUS server irrespective of its server reachability.
- `dead-only`: track the RADIUS server only when the server is marked as unreachable.

`vrf <VRF-NAME>`

Specifies the VRF name to be used for communicating with the server. If no VRF name is provided, the default VRF named `default` is used.

`spi <SPI-INDEX>`

Specifies the Security Parameters Index. The SPI is an identification tag carried in the IPsec AH header. The SPI must be unique on the switch. Range: 256 to 4294967295.

`<AUTH-TYPE>`

Specifies the authentication algorithm: `md5`, `sha1`, or `sha256`.

`<AUTH-KEY-TYPE>`

Specifies the authentication key type: `plaintext`, `hex-string`, or `ciphertext`.

[ `<AUTH-KEY>` ]

Specifies the authentication key. For `<AUTH-TYPE>` of `ciphertext`, this is the ciphertext string.

For `<AUTH-TYPE>` of `plaintext` or `hex-string`:

- md5 (plaintext): 1 to 16 characters, (hex-string): 2 to 32 hexadecimal digits.
- sha1 (plaintext): 1 to 20 characters, (hex-string): 2 to 40 hexadecimal digits.
- sha256 (plaintext): 1 to 32 characters, (hex-string): 2 to 64 hexadecimal digits.



When `<AUTH-KEY-TYPE>` is not followed by `<AUTH-KEY>`, plaintext authentication key prompting occurs upon pressing Enter. Enter must be pressed immediately after the `<AUTH-KEY-TYPE>` parameter without entering other parameters. The entered authentication key characters are masked with asterisks.

`<ENCRYPT-TYPE>`

Specifies the encryption algorithm: 3des, aes, des, or null.

`<ENCRYPT-KEY-TYPE>`

Specifies the encryption key type: plaintext, hex-string, or ciphertext.

[`<ENCRYPT-KEY>`]

Specifies the encryption key. For `<ENCRYPT-TYPE>` of ciphertext, this is the ciphertext string.

For `<ENCRYPT-TYPE>` of plaintext or hex-string:

- 3des (plaintext): 24 characters, (hex-string): 48 hexadecimal digits.
- aes (plaintext): 16, 24, or 32 characters, (hex-string): 32, 48, or 64 hexadecimal digits.
- des (plaintext): 8 characters, (hex-string): 16 hexadecimal digits.



When `<ENCRYPT-KEY-TYPE>` is not followed by `<ENCRYPT-KEY>`, plaintext encryption key prompting occurs upon pressing Enter. Enter must be pressed immediately after the `<ENCRYPT-KEY-TYPE>` parameter without entering other parameters. The entered encryption key characters are masked with asterisks.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

If the fully qualified domain name is provided for the RADIUS server host, a DNS server must be configured and accessible through the same VRF as mentioned for the server host. This configuration is required for the resolution of the RADIUS server hostname to its IP address. If a DNS server is not available for this VRF, the RADIUS servers reachable through this VRF must be configured by means of their IP addresses only.

## Examples

Adding a RADIUS server with an IPv4 address, a plaintext passkey, and IPsec authentication (md5 plaintext).

```
switch(config)# radius-server host 1.1.1.1 key plaintext 98ab vrf mgmt secure
ipsec authentication spi 261 md5 plaintext labc
```

Adding a RADIUS server with an IPv4 address and a prompted IPsec authentication (md5) plaintext authentication key.

```
switch(config)# radius-server host 1.1.1.1 secure ipsec authentication spi 261 md5
Enter the IPsec authentication key: *****
Re-Enter the IPsec authentication key: *****
```

Adding a RADIUS server with an IPv4 address, IPsec authentication (MD5 plaintext), and IPsec encryption (AES plaintext):

```
switch(config)# radius-server host 1.1.1.2 vrf mgmt secure
ipsec encryption spi 262 md5 plaintext 9xyz aes plaintext 1234567890abcdef
```

Adding a RADIUS server by providing an IPv4 address and IPsec MD5 authentication type, and then responding to prompts for the keys and encryption type:

```
switch(config)# radius-server host 1.1.1.6 secure ipsec encryption spi 262 md5
Enter the IPsec authentication key: *****
Re-Enter the IPsec authentication key: *****

Enter the IPsec encryption type (3des/aes/des/null)? aes

Enter the IPsec encryption key: *****
Re-Enter the IPsec encryption key: *****
```

Adding a RADIUS server with an IPv4 address, tracking enabled, tracking mode, IPsec authentication (MD5 plaintext), IPsec encryption (AES plaintext) is set to dead-only:

```
switch(config)# radius-server host 1.1.1.1 tracking enable tracking-mode dead-only
vrf mgmt secure ipsec encryption spi 262 md5 plaintext 9xyz
aes plaintext 1234567890abcdef
```

Removing a RADIUS server:

```
switch(config)# no radius-server host 1.1.1.1 vrf mgmt
```

Removing the ipsec configuration from a RADIUS server:

```
switch(config)# no radius-server host 1.1.1.2 vrf mgmt secure ipsec encryption
```

## radius-server host tls (RadSec)

### Syntax

```
radius-server host {<FQDN> | <IPV4> | <IPV6>}
tls [timeout <TIMEOUT-SECONDS>] [port <PORT-NUMBER>]
[auth-type {pap | chap}] [tracking {enable | disable}]
[tracking-mode {any | dead-only}] [vrf <VRF-NAME>]

no radius-server host {<FQDN> | <IPV4> | <IPV6>}
tls [port <PORT-NUMBER>] [vrf <VRF-NAME>]
```

### Description

Adds a RadSec server. By default, the RADIUS server is associated with the server group named `radius`. RadSec is used to secure the communication between RADIUS server and RADIUS client using TLS. The `no` form of this command removes a previously added RadSec server.



---

The shared key will be added as `radsec` for connection establishment.

---

### Command context

config

## Parameters

{<FQDN> | <IPv4> | <IPv6>}

Specifies the RADIUS server as:

- <FQDN>: a fully qualified domain name.
- <IPv4>: an IPv4 address.
- <IPv6>: an IPv6 address.

tls

Establishes RADIUS connection over TLS.

timeout <TIMEOUT-SECONDS>

Specifies the timeout. The range is 1 to 60 seconds. If a timeout is not specified, the value from the global timeout for RADIUS is used.

port <PORT-NUMBER>

Specifies the port number to establish RadSec connection. Range: 1 to 65535. Default: 2083.

auth-type {pap | chap}

Selects either PAP (default) or CHAP authentication type. If this parameter is not specified, the RADIUS global default is used.

tracking {enable | disable}

Enables or disables server tracking for the RADIUS server. Tracked servers are probed at the start of each server tracking interval to check if they are reachable.

Use command `radius-server tracking` to configure RADIUS server tracking globally.



---

Server tracking uses authentication request and response packets to determine server reachability status. The server tracking user name and password are used to form the request packet which is sent to the server with tracking enabled. Upon receiving a response to the request packet, the server is considered to be reachable.

---

tracking-mode {any | dead-only}

Configures tracking mode for the RADIUS server that has tracking enabled with the server. The tracking mode is used to monitor the status of RADIUS server reachability. The default tracking mode is `any`.

Sets the tracking mode to:

- `any`: track the RADIUS server irrespective of its server reachability.
- `dead-only`: track the RADIUS server only when the server is marked as unreachable.

vrf <VRF-NAME>

Specifies the VRF name to be used for communicating with the server. If no VRF name is provided, the default VRF named `default` is used.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Adding a RADIUS server over TLS with an IPv4 address and a named VRF:

```
switch(config)# radius-server host 1.1.1.1 tls vrf mgmt
```

Adding a RADIUS server over TLS with an IPv4 address and default port:

```
switch(config)# radius-server host 1.1.1.1 tls port
```

Adding a RADIUS server over TLS with tracking enabled and tracking mode is set to dead-only:

```
switch(config)# radius-server host 1.1.1.1 tls tracking enable tracking-mode dead-only
```

Adding a RADIUS server over TLS with an IPv4 address, a port, and a named VRF:

```
switch(config)# radius-server host 1.1.1.2 tls port 32 vrf mgmt
```

Adding a RADIUS server over TLS with an IPv6 address:

```
switch(config)# radius-server host 2001:0db8:85a3:0000:0000:8a2e:0370:7334 tls
```

## radius-server key

### Syntax

```
radius-server key [plaintext <GLOBAL-PASSKEY> | ciphertext <GLOBAL-PASSKEY>]
```

```
no radius-server key
```

### Description

Creates or modifies a RADIUS global passkey. The RADIUS global passkey is used as a shared-secret for encrypting the communication between all RADIUS servers and the switch. The RADIUS global passkey is required for authentication unless local passkeys have been set. By default, the RADIUS global passkey is empty. If the administrator has not set this key, the switch will not be able to perform RADIUS authentication. The switch will instead rely on the authentication mechanism configured with `aaa authentication login`.



---

When this command is entered without parameters, plaintext passkey prompting occurs upon pressing Enter. The entered passkey characters are masked with asterisks.

---

The `no` form of the command removes the global passkey.

### Command context

config

### Parameters

plaintext <GLOBAL-PASSKEY>

Specifies the RADIUS global passkey in plaintext format with a length of 1 to 31 characters. As per RFC 2865, a shared-secret can be a mix of alphanumeric and special characters.

ciphertext <GLOBAL-PASSKEY>

Specifies the RADIUS global passkey in encrypted format.

### Authority

Administrators or local user group members with execution rights for this command.

## Examples

Adding the global passkey:

```
switch(config)# radius-server key plaintext mypasskey123
```

Adding the global passkey with prompting:

```
switch(config)# radius-server key
Enter the RADIUS server key: *****
Re-Enter the RADIUS server key: *****
```

Removing the global passkey:

```
switch(config)# no radius-server key
```

## radius-server retries

### Syntax

```
radius-server retries <0-5>
no radius-server retries
```

### Description

Sets at the global level the number of retries the switch makes before concluding that the RADIUS server is unreachable.

You can override this setting with a fine-grained per RADIUS server retries configuration.

The `no` form of this command resets the RADIUS global retries to the default retries value of 1.

### Command context

config

### Parameters

retries <0-5>

Specifies the number of retry attempts for contacting RADIUS servers. Range is 0 to 5 retries.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Example

```
switch(config)# radius-server retries 3
```

## radius-server timeout

### Syntax

```
radius-server timeout [<1-60>]
no radius-server timeout
```

## Description

Specifies the number of seconds to wait for a response from the RADIUS server before trying the next RADIUS server. If a value is not specified, a default value of 5 seconds is used. You can override this value with a fine-grained per server timeout configured for individual servers.

The `no` form of this command resets the RADIUS global authentication timeout to the default of 5 seconds.

## Command context

config

## Parameters

timeout <1-60>

Specifies the timeout interval of 1 to 60 seconds. The default is 5 seconds.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Setting the RADIUS server timeout:

```
switch(config)# radius-server timeout 10
```

Resetting the timeout for the RADIUS server to the default:

```
switch(config)# no radius-server timeout
```

# radius-server tls timeout (RadSec)

## Syntax

```
radius-server tls timeout [<1-60>]
no radius-server tls timeout
```

## Description

Specifies the number of seconds to wait for a response from the RadSec server before trying the next RADIUS or RadSec server. If a value is not specified, a default value of 5 seconds is used. You can override this value with a fine-grained per server timeout configured for individual servers.

The `no` form of this command resets the RadSec global authentication timeout to the default of 5 seconds.

## Command context

config

## Parameters

timeout <1-60>

Specifies the timeout interval of 1 to 60 seconds. The default is 5 seconds.

## Authority



Administrators or local user group members with execution rights for this command.

## Examples

Setting the RadSec server timeout:

```
switch(config)# radius-server tls timeout 10
```

Resetting the timeout for the RadSec to the default:

```
switch(config)# no radius-server tls timeout
```

## radius-server tracking

### Syntax

```
radius-server tracking interval <INTERVAL>
no radius-server tracking interval

radius-server tracking retries <RETRIES>
no radius-server tracking retries

radius-server tracking user-name <NAME>
    [password [plaintext <PASSWORD> | ciphertext <PASSWORD>]]
no radius-server tracking user-name <NAME>
```

### Description

Configures RADIUS server tracking settings globally for all configured RADIUS servers that have tracking enabled with the `radius-server host` command on individual servers.

The `no` form of the command removes the specified configuration, reverting it to its default. The `no` form with `user-name` also clears the password (resets it to empty).

### Command context

config

### Parameters

interval <INTERVAL>

Specifies the time interval, in seconds, to wait before checking the server reachability status. Default: 300. Range 60 to 84600.

retries <RETRIES>

Specifies the number of server retries. Default: Global RADIUS retries. Range: 0 to 5.

user-name <NAME> [password [plaintext <PASSWORD> | ciphertext <PASSWORD>]]

Specifies the user name (and optionally a password) to be used for server checking. The default user name is `radius-tracking-user` with an empty password.

The password is optional and may be entered as `plaintext` or pasted in as `ciphertext`. The plaintext password is visible as cleartext when entered but is encrypted thereafter. Command history does show the password as cleartext.



When `password` is entered without a following sub-parameter, plaintext password prompting occurs upon pressing Enter. The entered password characters are masked with asterisks.



---

The user does not have to be configured on the server. Server tracking can still be performed with a user which is not configured on the server because authentication failure on the server achieves confirmation that the server is reachable.

---



---

Server tracking uses authentication request and response packets to determine server reachability status. The server tracking user name and password are used to form the request packet which is sent to the server with tracking enabled. Upon receiving a response to the request packet, the server is considered to be reachable.

---

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring a tracking interval of 120 seconds:

```
switch(config)# radius-server tracking interval 120
```

Reverting the tracking interval to its default of 300 seconds:

```
switch(config)# no radius-server tracking interval
```

Configuring three retries:

```
switch(config)# radius-server tracking retries 3
```

Configuring user `radius-tracker` with a plaintext password.

```
switch(config)# radius-server tracking user-name radius-tracker  
password plaintext track$1
```

Configuring user `radius-tracker` with a prompted plaintext password.

```
switch(config)# radius-server tracking user-name radius-tracker password  
Enter the RADIUS server tracking password: *****  
Re-Enter the RADIUS server tracking password: *****
```

Reverting the tracking user name to its default of `radius-tracking-user`:

```
switch(config)# no radius-server tracking user-name
```

## server

### Syntax

```
server {<FQDN> | <IPv4> | <IPv6>} [tls] [port <PORT-NUMBER>] [vrf <VRF-NAME>]  
no server {<FQDN> | <IPv4> | <IPv6>} [tls] [port <PORT-NUMBER>] [vrf <VRF-NAME>]
```

### Description

Adds a TACACS+/RADIUS server to a server-group. Only the configured TACACS+/RADIUS servers are allowed to be added within the server group. If the same server name exists with multiple ports or multiple VRFs, specify the server name, port, and VRF when adding the server to the server-group.

The `no` form of this command removes a TACACS+/RADIUS server from a server-group.

## Command context

`config-sg`

## Parameters

{<FQDN> | <IPv4> | <IPv6>}

Specifies the server as:

- <FQDN>: a fully qualified domain name.
- <IPv4>: an IPv4 address.
- <IPv6>: an IPv6 address.

`tls`

Specifies the TLS protection for the RADIUS server.

If TLS is configured without a port number, the system searches the RADIUS server by host name and sets the default authentication port (2083). Group server priority is assigned based on the sequence in which the servers are added.

`port <PORT-NUMBER>`

Specifies the authentication port number. Range: 1 to 65535. Default TACACS+ (TCP): 49, RADIUS (UDP): 1812 and RadSec: 2083.

If a port number is not provided, the system searches the TACACS+/RADIUS server by host name and sets the default authentication port. Group server priority is assigned based on the sequence in which the servers are added.

`vrf <VRF-NAME>`

Specifies the VRF name.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Adding a server to TACACS+ server group sg1 by providing an IPv4 address, port number, and VRF name:

```
switch(config)# aaa group server tacacs sg1
switch(config-sg)# server 1.1.1.2 port 32 vrf default
```

Adding a server to TACACS+ server group sg2 by providing an IPv6 address and default VRF:

```
switch(config)# aaa group server tacacs sg2
switch(config-sg)# server 2001:0db8:85a3:0000:0000:8a2e:0370:7334 vrf default
```

Adding a server to RADIUS server group sg3 by providing an IPv4 address, port number, and VRF name:

```
switch(config)# aaa group server radius sg3
switch(config-sg)# server 1.1.1.5 port 12 vrf default
```

Adding a server to RADIUS server group sg3 with TLS protection by providing an IPv4 address, port number, and VRF name:

```
switch(config)# aaa group server radius sg3
switch(config-sg)# server 1.1.1.5 tls port 12 vrf default
```

Adding a server to RADIUS server group sg4 by providing an IPv6 address and default VRF:

```
switch(config)# aaa group server radius sg4
switch(config-sg)# server 2001:0db8:85a3:0000:0000:8a2e:0371:7334 vrf default
```

Adding a server to RADIUS server group sg4 by providing an IPv4 address, port number, and VRF name:

```
switch(config)# aaa group server radius sg4
switch(config-sg)# server 1.1.1.6 port 32 vrf vrf_red
```

Specifying an IPv4 address when removing a TACACS+ server from server group sg1:

```
switch(config)# aaa group server tacacs sg1
switch(config-sg)# no server 1.1.1.2 port 12 vrf default
```

Specifying an IPv6 address when removing a TACACS+ server from server group sg2 with the default VRF:

```
switch(config)# aaa group server tacacs sg2
switch(config-sg)# no server 2001:0db8:85a3:0000:0000:8a2e:0370:7334 vrf default
```

## show aaa accounting

### Syntax

```
show aaa accounting [vsx-peer]
```

### Description

Shows the accounting configuration per connection type (channel).

### Command context

Operator (>) or Manager (#)

### Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Example

Configuring and then showing the accounting sequence for TACACS+ groups and local:

```
switch(config)# aaa accounting all default start-stop group tg1 tg2 tacacs local
switch(config)# aaa accounting all ssh start-stop group tg1 tg2
switch(config)# aaa accounting all console start-stop group tg4 tacacs local
switch(config)# aaa accounting all https-server start-stop local group tacacs tg3
switch(config)# exit
switch# show aaa accounting
AAA Accounting:
  Accounting Type           : all
  Accounting Mode           : start-stop

Accounting for default channel:
-----
GROUP NAME                  | GROUP PRIORITY
-----
tg1                          | 0
tg2                          | 1
tacacs                       | 2
local                        | 3
-----

Accounting for ssh channel:
-----
GROUP NAME                  | GROUP PRIORITY
-----
tg1                          | 0
tg2                          | 1
-----

Accounting for console channel:
-----
GROUP NAME                  | GROUP PRIORITY
-----
tg4                          | 0
tacacs                       | 1
local                        | 2
-----

Accounting for https-server channel:
-----
GROUP NAME                  | GROUP PRIORITY
-----
local                        | 0
tacacs                       | 1
tg3                          | 2
-----
```

Configuring and then showing the accounting sequence for RADIUS groups and local:

```
switch(config)# aaa accounting all default start-stop group rg1 rg2 radius local
switch(config)# aaa accounting all console start-stop group rg4 radius local
switch(config)# exit
switch# show aaa accounting
AAA Accounting:
  Accounting Type           : all
  Accounting Mode           : start-stop

Accounting for default channel:
-----
GROUP NAME                  | GROUP PRIORITY
-----
```

```

-----
rg1                               | 0
rg2                               | 1
radius                           | 2
local                            | 3
-----

```

Accounting for console channel:

```

-----
GROUP NAME                       | GROUP PRIORITY
-----
tg4                               | 0
radius                           | 1
local                            | 2
-----

```

Configuring and then showing only local accounting for default:

```

switch(config)# aaa accounting all default start-stop local
switch(config)# exit
switch# show aaa accounting
AAA Accounting:
  Accounting Type                : all
  Accounting Mode                : start-stop

Accounting for default channel:
-----
GROUP NAME                       | GROUP PRIORITY
-----
local                            | 0
-----

```

## show aaa authentication

### Syntax

```
show aaa authentication [vsx-peer]
```

### Description

Shows the authentication configuration per connection type (channel).

### Command context

Operator (>) or Manager (#)

### Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Example

Configuring TACACS+ authentication sequences and then showing the configuration per connection type (channel):

```
switch(config)# aaa authentication login default group tg1 tg2 tg3 tg4 tacacs local
switch(config)# aaa authentication login ssh group tg1 tg2
switch(config)# aaa authentication login console group tg4 tacacs local
switch(config)# aaa authentication login https-server local group tacacs tg3
switch(config)# exit
switch# show aaa authentication
AAA Authentication:
  Fail-through           : Enabled
  Limit Login Attempts   : Not set
  Lockout Time           : 300
  Minimum Password Length : Not set

Authentication for default channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
tg1                        | 0
tg2                        | 1
tg3                        | 2
tg4                        | 3
tacacs                     | 4
local                      | 5
-----

Authentication for ssh channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
tg1                        | 0
tg2                        | 1
-----

Authentication for console channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
tg4                        | 0
tacacs                     | 1
local                      | 2
-----

Authentication for https-server channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
local                      | 0
tacacs                     | 1
tg3                        | 2
-----
```

Configuring RADIUS authentication sequences and then showing the configuration per connection type (channel):

```
switch(config)# aaa authentication login default group rg1 rg2 rg3 rg4 radius local
switch(config)# aaa authentication login console group rg4 radius local
switch(config)# exit
switch# show aaa authentication
AAA Authentication:
```

```
Fail-through           : Enabled
Limit Login Attempts   : Not set
Lockout Time           : 300
Minimum Password Length : Not set
```

Authentication for default channel:

GROUP NAME	GROUP PRIORITY
rg1	0
rg2	1
rg3	2
rg4	3
radius	4
local	5

Authentication for console channel:

GROUP NAME	GROUP PRIORITY
rg4	0
radius	1
local	2

Configuring only default authentication and then showing the default connection type (channel):

```
switch(config)# aaa authentication login default local
switch(config)# exit
switch# show aaa authentication
```

```
AAA Authentication:
  Fail-through           : Disabled
  Limit Login Attempts   : Not set
  Lockout Time           : 300
  Minimum Password Length : Not set
```

Authentication for default channel:

GROUP NAME	GROUP PRIORITY
local	0

## show aaa authorization

### Syntax

```
show aaa authorization [vsx-peer]
```

### Description

Shows the authorization configuration per connection type (channel).

### Command context

Operator (>) or Manager (#)

### Parameters



[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

## Example

Configuring and then showing the authorization sequence for default and console connection types (channels):

```
switch(config)# aaa authorization commands default group tg1 tacacs none
All commands will fail if none of the servers in the group list are reachable.
Continue (y/n)? y
switch(config)#
switch(config)# aaa authorization commands console group tg1 tg2 tacacs none
All commands will fail if none of the servers in the group list are reachable.
Continue (y/n)? y
switch(config)# exit
switch#
switch# show aaa authorization
Authorization for default channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
tg1                        | 0
tacacs                     | 1
none                       | 2
-----

Authorization for console channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
tg1                        | 0
tg2                        | 1
tacacs                     | 2
none                       | 3
-----
```

## show aaa server-groups

### Syntax

```
show aaa server-groups [tacacs | radius] [vsx-peer]
```

### Description

Shows TACACS+ and RADIUS AAA server group information for all server types or for the specified server type.

### Command context

Operator (>) or Manager (#)

### Parameters

tacacs

Narrows the command output to only TACACS+ servers.

radius

Narrows the command output to only RADIUS servers.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Operators or Administrators or local user group members with execution rights for this command.  
Operators can execute this command from the operator context (>) only.

## Example

Showing all AAA server group information:

```
switch# show aaa server-groups

***** AAA Mechanism TACACS+ *****
-----
GROUP NAME          | SERVER NAME                               | PORT | VRF      | PRIORITY
-----
sg2                  | 2001:0db8:85a3:0000:0000:8a2e:          | 49   | default  | 1
                    | 0370:7334                               |
-----
sg1                  | 1.1.1.2                                 | 12   | mgmt     | 1
-----
tacacs (default)    | FQDN.com                               | 32   | mgmt     | 1
tacacs (default)    | 1.1.1.1                               | 49   | mgmt     | 2
tacacs (default)    | 1.1.1.2                               | 12   | mgmt     | 3
tacacs (default)    | abc.com                               | 32   | vrf_red  | 4
tacacs (default)    | 2001:0db8:85a3:0000:0000:8a2e:          | 49   | default  | 5
                    | 0370:7334                               |
tacacs (default)    | 1.1.1.3                               | 32   | vrf_blue | 6
-----
***** AAA Mechanism RADIUS *****
-----
GROUP NAME          | SERVER NAME                               | PORT | VRF      | PRIORITY
-----
sg4                  | 2001:0db8:85a3:0000:0000:8a2e:          | 1812 | default  | 1
                    | 0370:7334                               |
-----
sg3                  | 1.1.1.5                                 | 12   | mgmt     | 1
-----
radius (default)    | 1.1.1.4                               | 1812 | mgmt     | 1
radius (default)    | 1.1.1.5                               | 12   | mgmt     | 2
radius (default)    | abc1.com                               | 32   | mgmt     | 3
radius (default)    | 2001:0db8:85a3:0000:0000:8a2e:          | 1812 | default  | 4
                    | 0370:7334                               |
radius (default)    | 1.1.1.6                               | 32   | vrf_red  | 5
radius (default)    | 1.1.1.7                               | 32   | vrf_blue | 6
-----
```

Showing TACACS+ server group information:

```
switch# show aaa server-groups tacacs
```

```

***** AAA Mechanism TACACS+ *****
-----
GROUP NAME          | SERVER NAME                               | PORT | VRF      | PRIORITY
-----
sg2                  | 2001:0db8:85a3:0000:0000:8a2e:         | 49   | default  | 1
                    | 0370:7334                               |      |          |
-----
sg1                  | 1.1.1.2                                | 12   | mgmt     | 1
-----
tacacs (default)    | FQDN.com                               | 32   | mgmt     | 1
tacacs (default)    | 1.1.1.1                                | 49   | mgmt     | 2
tacacs (default)    | 1.1.1.2                                | 12   | mgmt     | 3
tacacs (default)    | abc.com                                 | 32   | vrf_red  | 4
tacacs (default)    | 2001:0db8:85a3:0000:0000:8a2e:         | 49   | default  | 5
                    | 0370:7334                               |      |          |
tacacs (default)    | 1.1.1.3                                | 32   | vrf_blue | 6
-----

```

Showing RADIUS server group information:

```

switch# show aaa server-groups radius

***** AAA Mechanism RADIUS *****
-----
GROUP NAME          | SERVER NAME                               | PORT | VRF      | PRIORITY
-----
sg4                  | 2001:0db8:85a3:0000:0000:8a2e:         | 1812 | default  | 1
                    | 0370:7334                               |      |          |
-----
sg3                  | 1.1.1.5                                | 12   | mgmt     | 1
-----
radius (default)    | 1.1.1.4                                | 1812 | mgmt     | 1
radius (default)    | 1.1.1.5                                | 12   | mgmt     | 2
radius (default)    | abc1.com                               | 32   | mgmt     | 3
radius (default)    | 2001:0db8:85a3:0000:0000:8a2e:         | 1812 | default  | 4
                    | 0370:7334                               |      |          |
radius (default)    | 1.1.1.6                                | 32   | vrf_red  | 5
radius (default)    | 1.1.1.7                                | 32   | vrf_blue | 6
-----

```

## show accounting log

### Syntax

```
show accounting log [last <QTY-TO-SHOW> | all]
```

### Description

Entered without optional parameters, this command shows all accounting log records for the current boot. Sensitive information is masked from the log, by being represented as asterisks.



This `show accounting log` command replaces the `show audit-log` command that is supported only in 10.00 releases.

### Command context

Manager (#) or Auditor (auditor>)

## Parameters

`last <QTY-TO-SHOW>`

Specifies how many most-recent accounting log records to show for the current boot. Range: 1 to 1000.

`all`

Selects for showing, all accounting records from the current boot and the previous boot.

## Authority

Auditors or Administrators or local user group members with execution rights for this command. Auditors can execute this command from the auditor context (auditor>) only.

## Usage

The log message starts with the record type, which is specific to AOS-CX. Values are the following:

`USER_START`

Record of a user login action.

`USER_END`

Record of a user logout action.

`USYS_CONFIG`

Record of a command executed by the user.

The three types of accounting log information are identified by the `msg=` element starting with the `rec=` item as follows:

- Exec is identified with: `msg='rec=ACCT_EXEC'`
- Command is identified with: `msg='rec=ACCT_CMD'`
- System is identified with: `msg='rec=ACCT_SYSTEM'`

The user group is indicated by `priv-lvl`, which is specific to AOS-CX. Values are the following:

Privilege level	User group
1	operators
15	administrators
19	auditors

The value of `service` indicates which user interface was used:

`service=shell`

Indicates that the log entry is a result of a CLI command.

`service=https-server`

Indicates that the log entry is a result of a REST API request or a Web UI action.

The string value of `data` identifies the CLI command or REST API request that was executed.

These elements are shown in context under *Examples*.

## Examples

Showing the accounting log for the previous and current boot. Line breaks have been added for readability.

```
switch# show accounting log all
```

-----  
Local accounting logs from previous boot  
-----

-----  
type=DAEMON\_START msg=audit(Nov 05 2018 23:00:58.607:9057) :  
auditd start, ver=2.4.3 format=raw kernel=4.9.119-yocto-standard res=success  
-----  
type=USER\_START msg=audit(Nov 05 2018 23:06:42.398:42) :  
msg='rec=ACCT\_EXEC op=start session=CONSOLE timezone=UTC user=user1 priv-lvl=15  
auth-method=LOCAL auth-type=LOCAL service=shell isconfig=no  
hostname=8xxx addr=0.0.0.0 res=success'  
-----  
type=USYS\_CONFIG msg=audit(Nov 05 2018 23:06:42.399:43) :  
msg='rec=ACCT\_CMD op=stop session=CONSOLE timezone=UTC user=user1 priv-lvl=15  
auth-method=LOCAL auth-type=LOCAL service=shell isconfig=no  
data="enable" hostname=8xxx addr=0.0.0.0 res=success'  
-----  
type=USYS\_CONFIG msg=audit(Nov 05 2018 23:08:24.693:51) :  
msg='rec=ACCT\_CMD op=stop session=CONSOLE timezone=UTC user=user1 priv-lvl=1  
auth-method=LOCAL auth-type=LOCAL service=shell isconfig=no  
data="configure terminal" hostname=8xxx addr=0.0.0.0 res=success'  
-----  
type=USYS\_CONFIG msg=audit(Nov 05 2018 23:08:39.108:52) :  
msg='rec=ACCT\_CMD op=stop session=CONSOLE timezone=UTC user=user1 priv-lvl=15  
auth-method=LOCAL auth-type=LOCAL service=shell isconfig=yes  
data="https-server rest access-mode read-write"  
hostname=8xxx addr=0.0.0.0 res=success'  
-----  
type=USER\_START msg=audit(Nov 05 2018 23:10:57.238:58) :  
msg='rec=ACCT\_EXEC op=start session=REST timezone=UTC user=admin priv-lvl=15  
auth-method=LOCAL auth-type=LOCAL service=https-server  
data="http-method=POST http-uri=/rest/v1/login"  
hostname=8xxx addr=127.0.0.1 res=success'  
-----  
type=USYS\_CONFIG msg=audit(Nov 05 2018 23:15:11.958:75) :  
msg='rec=ACCT\_CMD op=stop session=CONSOLE timezone=UTC user=user1 priv-lvl=15  
auth-method=LOCAL auth-type=LOCAL service=shell isconfig=yes  
data="tacacs-server host 2.2.2.2" hostname=8xxx addr=0.0.0.0 res=success'  
-----  
type=USYS\_CONFIG msg=audit(Nov 05 2018 23:15:37.090:76) :  
msg='rec=ACCT\_CMD op=stop session=REST timezone=UTC user=admin priv-lvl=15  
auth-method=LOCAL auth-type=LOCAL service=https-server  
data="http-method=GET http-uri=/rest/v1/system/vrfs/mgmt/tacacs\_servers"  
hostname=8xxx addr=127.0.0.1 res=success'  
-----  
type=USER\_END msg=audit(Nov 05 2018 23:26:59.207:90) :  
msg='rec=ACCT\_EXEC op=stop session=REST timezone=UTC user=admin priv-lvl=15  
auth-method=LOCAL auth-type=LOCAL service=https-server  
data="http-method=POST http-uri=/rest/v1/logout"  
hostname=8xxx addr=127.0.0.1 res=success'  
-----  
type=USER\_END msg=audit(Nov 05 2018 23:27:49.164:93) :  
msg='rec=ACCT\_EXEC op=stop session=CONSOLE timezone=UTC user=user1 priv-lvl=15  
auth-method=LOCAL auth-type=LOCAL service=shell isconfig=no  
hostname=8xxx addr=0.0.0.0 res=success'  
-----

-----  
Local accounting logs from current boot  
-----

-----  
type=DAEMON\_START msg=audit(Nov 05 2018 23:32:05.642:626) :  
auditd start, ver=2.4.3 format=raw kernel=4.9.119-yocto-standard res=success  
-----

```
-----
type=USER_START msg=audit(Nov 05 2018 23:35:52.915:11) :
msg='rec=ACCT_EXEC op=start session=CONSOLE timezone=UTC user=admin priv-lvl=15
auth-method=LOCAL auth-type=LOCAL service=shell isconfig=no
hostname=8xxx addr=0.0.0.0 res=success'
-----
type=USYS_CONFIG msg=audit(Nov 05 2018 23:35:52.917:12) :
msg='rec=ACCT_CMD op=stop session=CONSOLE timezone=UTC user=admin priv-lvl=15
auth-method=LOCAL auth-type=LOCAL service=shell isconfig=no data="enable"
hostname=8xxx addr=0.0.0.0 res=success'
```

## show radius-server

### Syntax

```
show radius-server [detail] [vsx-peer]
```

### Description

Shows configured RADIUS servers information.

### Command context

Operator (>) or Manager (#)

### Parameters

**detail**

Selects additional RADIUS server details and global parameters for showing.

**[vsx-peer]**

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Usage

When the `show radius-server` command shows `None` for the shared-secret, the passkey is missing.

### Examples

Showing a summary of the global RADIUS configuration:

```
switch# show radius-server
***** Global RADIUS Configuration *****

Shared-Secret: None
Timeout: 60
Auth-Type: pap
Retries: 5
TLS Timeout: 60
Tracking Time Interval (seconds): 60
Tracking Retries: 5
Tracking User-name: radius-tracking-user
```

```
Tracking Password: None
Number of Servers: 1
```

```
-----
SERVER NAME          | TLS | PORT | VRF
-----
20.1.1.129           |    | 1812 | default
1.1.1.4               |    | 1812 | mgmt
1.1.1.5               |    | 12   | mgmt
abc1.com              |    | 32   | mgmt
2001:0db8:85a3:0000:0000:8a2e:0371:7334 |    | 1812 | default
1.1.1.6               |    | 32   | vrf_red
1.1.1.7               |    | 32   | vrf_blue
-----
```

Showing a summary of a RADIUS server when the per-server shared key or the global RADIUS shared key is not set:

```
switch# show radius-server
***** Global RADIUS Configuration *****
```

```
Shared-Secret: None
Timeout: 60
Auth-Type: pap
Retries: 5
TLS Timeout: 60
Tracking Time Interval (seconds): 60
Tracking Retries: 5
Tracking User-name: radius-tracking-user
Tracking Password: None
Number of Servers: 1
```

```
-----
SERVER NAME          | TLS | PORT | VRF
-----
20.1.1.129           |    | 1812 | default
-----
```

Showing details of a global RADIUS configuration:

```
switch# show radius-server detail
***** Global RADIUS Configuration *****

Shared-Secret: AQBapb+HsdpqVlQcA+CyD0RvfbeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout: 5
Auth-Type: pap
Retries: 5
TLS Timeout: 60
Tracking Time Interval (seconds): 60
Tracking Retries: 5
Tracking User-name: radius-tracking-user
Tracking Password: None
Number of Servers: 1

***** RADIUS Server Information *****
Server-Name          : 20.1.1.129
Auth-Port             : 1812
Accounting-Port       : 1813
VRF                   : default
```

```

TLS Enabled      : No
Shared-Secret    : None
Timeout          : 60
Retries          : 5
Auth-Type        : pap
Server-Group     : radius
Default-Priority : 4
Tracking         : disabled
Tracking-Mode     : any
Reachability-Status : N/A
ClearPass-Username : 
ClearPass-Password : None

```

Showing details of a RADIUS server when the per-server shared key and the global RADIUS shared key are not set:

```

switch# show radius-server detail
***** Global RADIUS Configuration *****

Shared-Secret: None
Timeout: 5
Auth-Type: pap
Retries: 1
TLS Timeout: 5
Number of Servers: 1

***** RADIUS Server Information *****
Server-Name      : 1.1.1.1
Auth-Port        : 2083
VRF              : default
Shared-Secret (default) : None
Timeout (default) : 5
Retries (default) : 1
Auth-Type (default) : pap
Server-Group (default) : radius
Default-Priority : 1

```

Showing details of a RADIUS server with TLS:

```

switch# show radius-server detail
***** Global RADIUS Configuration *****

Shared-Secret: None
Timeout: 5
Auth-Type: pap
Retries: 1
TLS Timeout: 5
TLS Connection Timeout: 5
TLS Connection Retries: 1
Tracking Time Interval (seconds): 60
Tracking Retries: 1
Tracking User-name: jim
Tracking Password:
AQBapcPi5GR2MGH5WCuYW0ZRtLrFgGT/pAcp0JqFFpndkMwYCwAAADsYv787vMdnbSBZ
Number of Servers: 1

***** RADIUS Server Information *****
Server-Name      : 172.20.30.30
Auth-Port        : 2083

```



```

Accounting-Port      : 2083
VRF                  : default
TLS Enabled          : Yes
TLS Connection Timeout (default): 5
TLS Connection Retries (default): 1
TLS Connection Status : tls_connection_established
Timeout (default)    : 5
Auth-Type (default)  : pap
Server-Group (default) : radius
Default-Priority      : 1
Tracking              : enabled
Tracking-Mode         : any
Reachability-Status   : reachable
ClearPass-Username    : admin
ClearPass-Password    :
AQBapcPi5GR2MGH5WCuYW0ZRtLrFgGT/pAcp0JqFFpndkMwYCwAAADsYv787vMdnbSBZ

```

## show radius-server secure ipsec

### Syntax

```
show radius-server secure ipsec { server-list | host {<FQDN> | <IPv4> | <IPv6>} [port <PORT-NUMBER>] [vrf <VRF-NAME>] [vsx-peer] }
```

### Description

Shows information for one or all RADIUS servers configured with IPsec.

### Command context

Operator (>) or Manager (#)

### Parameters

**server-list**

Selects all servers for showing.

**host {<FQDN> | <IPv4> | <IPv6>}**

Specifies the RADIUS server as:

- <FQDN>: a fully qualified domain name.
- <IPv4>: an IPv4 address.
- <IPv6>: an IPv6 address.

**port <PORT-NUMBER>**

Specifies the authentication port number. Range: 1 to 65535. Default: 1812.

**vrf <VRF-NAME>**

Specifies the VRF name to be used for communicating with the server. If no VRF name is provided, the default VRF named `default` is used.

**[vsx-peer]**

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

## Usage

The IPsec key is shown in an exportable ciphertext format.

## Examples

Showing information for RADIUS server 1.1.1.1 secured with IPsec:

```
switch# show radius-server secure ipsec host 1.1.1.1
IPsec           : enabled
Protocol        : ESP
Authentication   : MD5
Encryption      : AES
SPI             : 1234
```

Showing information for all RADIUS servers secured with IPsec:

```
switch# show radius-server secure ipsec server-list
Server          : 1.1.1.1
IPsec           : enabled
Protocol        : ESP
Authentication   : MD5
Encryption      : AES
SPI             : 1234

Server          : 1.1.1.2
IPsec           : enabled
Protocol        : ESP
Authentication   : MD5
Encryption      : AES
SPI             : 12341
```

## show radius-server statistics

### Syntax

```
show radius-server statistics authentication [vsx-peer]
```

### Description

Shows authentication statistics for all configured RADIUS servers.

### Command context

Operator (>) or Manager (#)

### Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

## Examples

Showing RADIUS server authentication statistics:

```
switch# show radius-server statistics authentication
Server Name      : rad1
Auth-Port        : 1812
Accounting-Port  : 1813
VRF              : mgmt
TLS Enabled      : Yes

Authentication Statistics
-----
Round Trip Time      : 100
Pending Requests     : 0
Timeouts             : 6
Bad Authenticators   : 2
Packets Dropped      : 0
Access Requests      : 20
Access Challenge     : 8
Access Accepts       : 14
Access Rejects       : 0
Access Response Malformed : 0
Access Retransmits   : 0
Tracking Requests    : 5
Tracking Responses   : 5
Unknown Response Code : 0
```

## show radius-server statistics host

### Syntax

```
show radius-server statistics authentication host {<FQDN> | <IPv4> | <IPv6>}
[tls] [port <PORT-NUMBER>] [vrf <VRF-NAME>] [vsx-peer]
```

### Description

Shows authentication statistics for the specified RADIUS server on the specified vrf.

### Command context

Operator (>) or Manager (#)

### Parameters

*authentication*

Selects authentication statistics to show.

*host* {<FQDN> | <IPv4> | <IPv6>}

Specifies the RADIUS server as:

- <FQDN>: a fully qualified domain name.
- <IPv4>: an IPv4 address.
- <IPv6>: an IPv6 address.

*tls*

Establishes RADIUS connection over TLS.

*port* <PORT-NUMBER>

Specifies the authenticated port. Range: 1 to 65535.

*vrf* <VRF-NAME>

Specifies the VRF name to be used for communicating with the server. If no VRF name is provided, the default VRF named `default` is used.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

## Examples

Showing RADIUS server authentication statistics with TLS enabled:

```
switch# show radius-server statistics authentication host 20.1.1.49 tls
Server Name      : 20.1.1.49
Auth-Port       : 2083
Accounting-Port  : 2083
VRF              : default
TLS Enabled      : Yes

Authentication Statistics
-----
Round Trip Time   : 3
Pending Requests  : 0
Timeouts          : 0
Bad Authenticators : 0
Packets Dropped   : 0
Access Requests   : 13
Access challenge   : 6
Access Accepts     : 3
Access Rejects     : 4
Access Response Malformed : 0
```

# show tacacs-server

## Syntax

```
show tacacs-server [detail] [vsx-peer]
```

## Description

Shows the configured TACACS+ servers.

## Command context

Operator (>) or Manager (#)

## Parameters

`detail`

Selects additional TACACS+ server details and global parameters for showing.

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Operators or Administrators or local user group members with execution rights for this command.  
Operators can execute this command from the operator context (>) only.

## Examples

Showing a summary of a global TACACS+ configuration with a shared-secret:

```
switch# show tacacs-server
***** Global TACACS+ Configuration *****

Shared-Secret: AQBapb+HsdpqV1Q3CPCBMQTG8e1cA+CyD0RvfbeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout: 5
Auth-Type: pap
Number of Servers: 5

-----
SERVER NAME                               | PORT | VRF
-----
1.1.1.1                                  | 49    | mgmt
1.1.1.2                                  | 12    | mgmt
abc.com                                  | 32    | vrf_blue
2001:0db8:85a3:0000:0000:8a2e:0370:7334 | 49    | default
1.1.1.3                                  | 32    | vrf_red
-----
```

Showing details of a global TACACS+ configuration:

```
switch# show tacacs-server detail
***** Global TACACS+ Configuration *****

Shared-Secret: AQBapb+HsdpqV1Q3CPCBMQTG8e1cA+CyD0RvfbeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout: 5
Auth-Type: pap
Number of Servers: 5

***** TACACS+ Server Information *****
Server-Name       : 1.1.1.2
Auth-Port         : 12
VRF               : mgmt
Shared-Secret (default) : AQBapb+HsdpqV1Q3CPCBMQTG8eeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout (default)  : 5
Auth-Type (default) : pap
Server-Group      : sg1
Group-Priority    : 1

Server-Name       : 2001:0db8:85a3:0000:0000:8a2e:0370:7334
Auth-Port         : 49
VRF               : default
Shared-Secret (default) : AQBapb+HsdpqV1Q3CPCBMQTG8eeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout (default)  : 5
Auth-Type (default) : pap
Server-Group      : sg2
Group-Priority    : 1

Server-Name       : 1.1.1.1
Auth-Port         : 49
VRF               : mgmt
Shared-Secret (default) : AQBapb+HsdpqV1Q3CPCBMQTG8eeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout (default)  : 5
```

```

Auth-Type (default)      : pap
Server-Group (default)   : tacacs
Default-Priority         : 1

Server-Name              : abc.com
Auth-Port                : 32
VRF                      : vrf_red
Shared-Secret (default) : AQBapb+HsdpqV1Q3CPCBMQTG8eeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout                  : 15
Auth-Type (default)     : pap
Server-Group (default)   : tacacs
Default-Priority         : 3

Server-Name              : 1.1.1.3
Auth-Port                : 32
VRF                      : vrf_blue
Shared-Secret            : AQBapfnqbSswqKC476tdUFZ+AncIRY92hDTYkQCAAAAFEaHn43vNC
Timeout                  : 15
Auth-Type                : chap
Server-Group (default)   : tacacs
Default-Priority         : 5

```

Showing TACACS+ server when per-server shared key and global TACACS+ shared key is not set:

```

switch# show tacacs-server
***** Global TACACS+ Configuration *****

Shared-Secret: None
Timeout: 5
Auth-Type: pap
Number of Servers: 1

-----
SERVER NAME                               | PORT | VRF
-----
1.1.1.1                                  | 49   | default
-----

```

Showing TACACS+ server details when per-server shared key and global TACACS+ shared key is not set:

```

switch# show tacacs-server detail
***** Global TACACS+ Configuration *****

Shared-Secret: None
Timeout: 5
Auth-Type: pap
Number of Servers: 1

***** TACACS+ Server Information *****
Server-Name      : 1.1.1.1
Auth-Port        : 49
VRF              : default
Shared-Secret (default) : None
Timeout (default)   : 5
Auth-Type (default) : pap
Server-Group (default) : tacacs
Default-Priority   : 1

```

## show tacacs-server statistics

## Syntax

```
show tacacs-server statistics [vsx-peer]
```

## Description

Shows authentication statistics for all configured TACACS+ servers.

## Command context

Operator (>) or Manager (#)

## Parameters

[vsx-peer]

Shows the output from the VSX peer switch. If the switches do not have the VSX configuration or the ISL is down, the output from the VSX peer switch is not displayed. This parameter is available on switches that support VSX.

## Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

## Examples

Showing TACACS+ server authentication statistics:

```
switch# show tacacs-server statistics
Server Name      : tacl
Auth-Port       : 49
VRF             : mgmt
Authentication Statistics
-----
Round Trip Time      : 1
Pending Requests    : 0
Timeout            : 0
Unknown Types       : 0
Packet Dropped      : 0
Auth Start         : 8
Auth challenge      : 0
Auth Accepts       : 4
Auth Rejects       : 4
Auth reply malformed : 0
Tracking Requests   : 0
Tracking Responses  : 0
```

# show tech aaa

## Syntax

```
show tech aaa
```

## Description

Shows the AAA configuration settings.

## Command context

Manager (#)

## Authority

Administrators or local user group members with execution rights for this command.

## Example

Showing the AAA configuration settings:

```
switch# show tech aaa
=====
Show Tech executed on Tue Feb 14 02:19:11 2017
=====
[Begin] Feature aaa
=====

*****
Command : show aaa authentication
*****
AAA Authentication:
  Fail-through           : Enabled
  Limit Login Attempts   : Not set
  Lockout Time           : 300
  Minimum Password Length : Not set

Authentication for ssh channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
local                      | 0
-----

Authentication for https-server channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
local                      | 0
-----

Authentication for console channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
local                      | 0
-----

Authentication for default channel:
-----
GROUP NAME                | GROUP PRIORITY
-----
tacacs                    | 0
local                     | 1
-----

*****
Command : show aaa accounting
*****
AAA Accounting:
  Accounting Type         : all
  Accounting Mode         : start-stop

Accounting for default channel:
-----
```



GROUP NAME	GROUP PRIORITY
------------	----------------

local	0
-------	---

Accounting for ssh channel:

GROUP NAME	GROUP PRIORITY
------------	----------------

tacacs	0
local	1

Accounting for https-server channel:

GROUP NAME	GROUP PRIORITY
------------	----------------

tacacs	0
--------	---

\*\*\*\*\*

Command : show aaa authorization

\*\*\*\*\*

Authorization for default channel:

GROUP NAME	GROUP PRIORITY
------------	----------------

none	0
------	---

Authorization for console channel:

GROUP NAME	GROUP PRIORITY
------------	----------------

none	0
------	---

Authorization for ssh channel:

GROUP NAME	GROUP PRIORITY
------------	----------------

tacacs	0
none	1

\*\*\*\*\*

Command : show aaa server-groups

\*\*\*\*\*

\*\*\*\*\* AAA Mechanism TACACS+ \*\*\*\*\*

GROUP NAME	SERVER NAME	PORT	PRIORITY	VRF
tacacs (default)	1.1.1.1	49	1	mgmt

\*\*\*\*\* AAA Mechanism RADIUS \*\*\*\*\*

GROUP NAME	SERVER NAME	PORT	PRIORITY	VRF
------------	-------------	------	----------	-----

```

*****
Command : show tacacs-server detail
*****
***** Global TACACS+ Configuration *****

Shared-Secret: G8ekK1cA+CyD0RvfbeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout: 5
Auth-Type: pap
Tracking: disabled
Tracking Time Interval (seconds): 300
Tracking User-name: tacacs-tracking-user
Tracking Password: None
Number of Servers: 1

***** TACACS+ Server Information *****
Server-Name           : 1.1.1.1
Auth-Port             : 49
VRF                   : mgmt
Shared-Secret         : KCdmOMxMD26T0fQoXfJbtj9j2AUxlGn6eCAAAAF2MkfMTtojQX
Timeout (default)     : 5
Auth-Type (default)   : pap
Server-Group (default): tacacs
Default-Priority      : 1
Tracking              : disabled
Reachability-Status   : N/A

*****
Command : show radius-server detail
*****
***** Global RADIUS Configuration *****

Shared-Secret: CPCBMQTG8ekK1cA+CyD0RvfbeA8BEgikCgAAAJOWZSNzA2SWrLA=
Timeout: 5
Auth-Type: pap
Retries: 1
Number of Servers: 0

=====
[End] Feature aaa
=====

=====
Show Tech commands executed successfully
=====

```

## tacacs-server auth-type

### Syntax

```

tacacs-server auth-type {pap | chap}
no tacacs-server auth-type

```

### Description

Enables the CHAP or PAP authentication protocol, which is used for communication with the TACACS+ servers, at the global level. You can override this command with a fine-grained per server `auth-type` configuration.

The `no` form of this command resets the global authentication mechanism for TACACS+ to PAP, which is the default authentication mechanism for TACACS+.

## Command context

config

## Parameters

auth-type {pap | chap}

Selects either the PAP or CHAP authentication protocol.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Enabling command for CHAP authentication:

```
switch(config)# tacacs-server auth-type chap
```

Enabling command for PAP authentication:

```
switch(config)# tacacs-server auth-type pap
```

# tacacs-server host

## Syntax

```
tacacs-server host {<FQDN> | <IPV4> | <IPV6>}  
    [key [plaintext <PASSKEY> | ciphertext <PASSKEY>]]  
    [timeout <TIMEOUT-SECONDS>] [port <PORT-NUMBER>]  
    [auth-type {pap | chap}] [tracking {enable | disable}] [vrf <VRF-NAME>]
```

```
no tacacs-server host {<FQDN> | <IPV4> | <IPV6>}  
    [port <PORT-NUMBER>] [vrf <VRF-NAME>]
```

## Description

Adds a TACACS+ server. By default, the TACACS+ server is associated with the server group named `tacacs`. The `no` form of this command removes a previously added TACACS+ server.

## Command context

config

## Parameters

{<FQDN> | <IPV4> | <IPv6>}

Specifies the TACACS+ server as:

- <FQDN>: a fully qualified domain name.
- <IPV4>: an IPv4 address.
- <IPV6>: an IPv6 address.

key {plaintext <PASSKEY> | ciphertext <PASSKEY>}

Specifies either a plaintext or an encrypted local shared-secret passkey for the server. As per RFC 2865, shared-secret can be a mix of alphanumeric and special characters. The length of shared-secret in plaintext format is fewer than 32 characters.



When `key` is entered without either sub-parameter, plaintext passkey prompting occurs upon pressing Enter. The entered passkey characters are masked with asterisks.

When `key` is omitted, the server uses the global passkey. This command requires either the global or local passkey to be set; otherwise the server will not be contacted. Command `radius-server key` is available for setting the global passkey.

`timeout <TIMEOUT-SECONDS>`

Specifies the timeout. The range is 1 to 60 seconds. The default timeout is 5 seconds.

`port <PORT-NUMBER>`

Specifies the TCP authentication port number. Range: 1 to 65535. Default: 49.

`auth-type {pap | chap}`

Selects either PAP (default) or CHAP authentication type. If this parameter is not specified, the TACACS+ global default is used.

`tracking {enable | disable}`

Enables or disables server tracking for the server. Tracked servers are probed at the start of each server tracking interval to check if they are reachable. Unreachable servers are skipped in favor of servers that are proven to be reachable. Use command `tacacs-server tracking` to configure TACACS+ server tracking.

`vrf <VRF-NAME>`

Specifies the VRF name to be used for communicating with the server. If no VRF name is provided, the default VRF named `default` is used.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

If the fully qualified domain name is provided for the TACACS+ server, a DNS server must be configured and accessible through the same VRF which is configured for the TACACS+ server. This configuration is required for the resolution of the TACACS+ server hostname to its IP address. If a DNS server is not available for this VRF, the TACACS+ servers reachable through this VRF must be configured by means of their IP addresses only.

## Examples

Adding a TACACS+ server with an IPv4 address, plaintext passkey, timeout, port, authentication type, and VRF name:

```
switch(config)# tacacs-server host 1.1.1.3 key plaintext test-123 timeout 15 port 32
auth-type chap vrf vrf_red
```

Adding a TACACS+ server with an IPv4 address and prompted plaintext passkey:

```
switch(config)# tacacs-server host 1.1.1.5 key
Enter the TACACS server key: *****
Re-Enter the TACACS server key: *****
```

Adding a TACACS+ server with an IPv4 address and a named VRF:

```
switch(config)# tacacs-server host 1.1.1.1 vrf mgmt
```

Adding a TACACS+ server with an IPv4 address, a port, and a named VRF:

```
switch(config)# tacacs-server host 1.1.1.2 port 32 vrf mgmt
```

Adding a TACACS+ server with an FQDN, a timeout, port number, and a named VRF:

```
switch(config)# tacacs-server host abc.com timeout 15 port 32 vrf vrf_blue
```

Adding a TACACS+ server with an IPv6 address:

```
switch(config)# tacacs-server host 2001:0db8:85a3:0000:0000:8a2e:0370:7334
```

Deleting a TACACS+ server with an IPv4 address and specified VRF:

```
switch(config)# no tacacs-server host 1.1.1.1 vrf mgmt
```

Deleting a TACACS+ server with an FQDN, port, and specified VRF:

```
switch(config)# no tacacs-server host abc.com port 32 vrf vrf_blue
```

## tacacs-server key

### Syntax

```
tacacs-server key [plaintext <GLOBAL-PASSKEY> | ciphertext <GLOBAL-PASSKEY>]
```

```
no tacacs-server key
```

### Description

Creates or modifies a TACACS+ global passkey. The TACACS+ global passkey is used as a shared-secret for encrypting the communication between all TACACS+ servers and the switch. The TACACS+ global passkey is required for authentication unless local passkeys have been set. By default, the TACACS+ global passkey is empty. If the administrator has not set this key, the switch will not be able to perform TACACS+ authentication. The switch will instead rely on the authentication mechanism configured with `aaa authentication login`.



---

When this command is entered without parameters, plaintext passkey prompting occurs upon pressing Enter. The entered passkey characters are masked with asterisks.

---

The `no` form of the command removes the global passkey.

### Command context

config

### Parameters

plaintext <GLOBAL-PASSKEY>

Specifies the TACACS+ global passkey in plaintext format with a length of 1 to 31 characters. As per RFC 2865, shared-secret can be a mix of alphanumeric and special characters.

ciphertext <GLOBAL-PASSKEY>

Specifies the TACACS+ global passkey in encrypted format.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Adding the global passkey:

```
switch(config)# tacacs-server key plaintext mypasskey123
```

Adding the global passkey with prompting:

```
switch(config)# tacacs-server key
Enter the TACACS server key: *****
Re-Enter the TACACS server key: *****
```

Removing the global passkey:

```
switch(config)# no tacacs-server key
```

## tacacs-server timeout

### Syntax

```
tacacs-server timeout [<1-60>]
no tacacs-server timeout
```

### Description

Specifies the number of seconds to wait for a response from the TACACS+ server before trying the next TACACS+ server. If a value is not specified, a default value of 5 seconds is used. You can override this value with a fine-grained per server timeout configured for individual servers.

The `no` form of this command resets the TACACS+ global authentication timeout to the default of 5 seconds.

### Command context

config

### Parameters

timeout <1-60>

Specifies the timeout interval of 1 to 60 seconds. The default is 5 seconds.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Specifying the TACACS+ server timeout:

```
switch(config)# tacacs-server timeout 10
```

Resetting the timeout for the TACACS+ server to the default:

```
switch(config)# no tacacs-server timeout
```

## tacacs-server tracking

### Syntax

```
tacacs-server tracking interval <INTERVAL>
no tacacs-server tracking interval

tacacs-server tracking user-name <NAME>
    [password [plaintext <PASSWORD> | ciphertext <PASSWORD>]]
no tacacs-server tracking user-name <NAME>
```

### Description

Configures TACACS+ server tracking settings globally for all configured TACACS+ servers that have tracking enabled with the `tacacs-server host` command on individual servers.

The `no` form of the command removes the specified configuration, reverting it to its default. The `no` form with `user-name` also clears the password (resets it to empty).

### Command context

config

### Parameters

interval <INTERVAL>

Specifies the time interval, in seconds, to wait before checking the server reachability status. Default: 300. Range 60 to 84600.

user-name <NAME> [password [plaintext <PASSWORD> | ciphertext <PASSWORD>]]

Specifies the user name (and optionally a password) to be used for server checking. The default user name is `tacacs-tracking-user` with an empty password.

The password is optional and may be entered as `plaintext` or pasted in as `ciphertext`. The plaintext password is visible as cleartext when entered but is encrypted thereafter. Command history does show the password as cleartext.



---

When `password` is entered without a following sub-parameter, plaintext password prompting occurs upon pressing Enter. The entered password characters are masked with asterisks.

---



---

The user does not have to be configured on the server. Server tracking can still be performed with a user which is not configured on the server because authentication failure on the server achieves confirmation that the server is reachable.

---



---

Server tracking uses authentication request and response packets to determine server reachability status. The server tracking user name and password are used to form the request packet which is sent to the server with tracking enabled. Upon receiving a response to the request packet, the server is considered to be reachable.

---

### Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring a tracking interval of 120 seconds:

```
switch(config)# tacacs-server tracking interval 120
```

Reverting the tracking interval to its default of 300 seconds:

```
switch(config)# no tacacs-server tracking interval
```

Configuring user `tacacs-tracker` with a plaintext password.

```
switch(config)# tacacs-server tracking user-name tacacs-tracker password plaintext track$1
```

Configuring user `tacacs-tracker` with a prompted plaintext password.

```
switch(config)# tacacs-server tracking user-name tacacs-tracker password  
Enter the TACACS server tracking password: *****  
Re-Enter the TACACS server tracking password: *****
```

Reverting the tracking user name to its default of `tacacs-tracking-user`:

```
switch(config)# no tacacs-server tracking user-name
```



The public key infrastructure (PKI) feature enables administrators to manage digital certificates on the switch. The switch uses certificates to validate SSH clients when acting as an SSH server, and when communicating with syslog servers when TLS encryption is used.

## PKI concepts

### Digital certificate

A digital certificate is an electronic form of identification that stores important information about an entity (such as a computer, program, or website). Certificates help secure digital transactions by enabling the end parties to validate each other's identity. Digital certificates are issued by a certificate authority (CA) and are composed of an encoded string of characters (usually stored in a file). For example:

```
-----BEGIN CERTIFICATE-----
MIIDSDCCApGCCQDJotuPPj9GCDANBgkqhkiG9w0BAQsAADCBCqzELMAkGA1UEBh
VVMxEzARBgNVBAGMCkNhbGlm3JuaWEExEDAOBgNVBACBM1JvY2tsaW4xDDAKBg
BAoMA0hQTjEVMBMGAlUECwwMSFBOUm9zZXZpbGx1MSokwAYDVQQDDCFocG5zd
...
MioDy0096DvSMPsnOaI+jnZ3AozN8y+nLgotXUsg36pO/Ncc51oQhyUdcAbgA1
rzSLgyTnpXZKumvlaoTk3pZrIf7m5V103GTbgHGSFCzgO6QWxVxu9d7ju1o59S
aOIT7JSsYI5LsLpVz9ZqS599rj/1LoH+rLN1RDVXpS+J51
-----END CERTIFICATE-----
```

The switch can import PEM encoded ITU-T X.509 v3 certificates. (Certificates can be converted to human-readable form using a software decoder.)

An X.509 digital certificate typically includes the following information:

- Signature algorithm: The cryptographic algorithm used to generate the digital signature.
- Signature value: Digital signature of the certificate generated using the CA's private key.
- Version number: X.509 version number.
- Serial number: Certificate serial number.
- Issuer name: Name of the certificate authority (CA) that issued the certificate.
- Validity period: Beginning and ending dates.
- Subject name: Name of the entity to which the certificate is issued.
- Subject public key and key algorithm.
- Key usage extension: Purpose of the certificate.

### Certificate authority

A certificate authority (CA) is an entity that can issue and sign digital certificates. A CA can be a well-known, trusted commercial company, or a private entity controlled by your organization. For a commercial CA, the CA validates the credentials of a user before issuing a certificate and signing it, guaranteeing a certificate holder's identity. For a private CA, self-signed certificates can be generated as needed for devices on your network without paying a commercial company.

### Root certificate

A root certificate is a self-signed certificate that is deemed the root of trust for a certificate chain. This is the certificate that identifies a CA, and is used by the CA to sign any certificates that it issues. When two peers attempt to establish a secure connection, they use the CA's public key to verify that each other's certificates were indeed signed by a trusted certificate authority.

Each root CA certificate has a unique fingerprint, which is the hash value of the certificate content. The fingerprint of a root CA certificate can be used to authenticate the validity of the root CA.

In a certificate chain, the root CA generates a self-signed certificate, and each lower level CA holds a CA certificate (intermediate certificate) issued by the CA immediately above it. The hierarchy of these certificates forms a *chain of trust*.

## Leaf certificate

This is the certificate used by a software entity, such as a syslog client, to identify itself to a peer when establishing a secure connection.

## Intermediate certificate

An intermediate certificate is a CA which has been issued by the root certificate or by another intermediate certificate. Intermediate CAs can issue leaf certificates and sit in between the root and leaf certificates. The use of an intermediate CA allows administrators to segregate their PKI groups.

## Trust anchor

This is the certificate that acts as the base of trust for the validation of other certificates. A trust anchor can be a root or intermediate certificate issued by a CA.

## OCSP

The online certificate status protocol (OCSP) is a real-time method for determining the revocation status of a certificate. When two peers attempt to establish a secure connection, they can query an OCSP responder to determine the status (valid or revoked) of each other's certificates. The OCSP responder for a certificate is typically provided by a server managed by the CA that issued the certificate.

## PKI on the switch

The AOS-CX Switch Series switches provides for installation of certificate authority (CA) certificates and the generation and installation of leaf certificates.

## Trust anchor profiles

The switch supports 64 trust anchor (TA) profiles. Each TA profile stores a trusted CA certificate. The certificate can be either a root CA certificate, which must be self-signed, or an intermediate CA certificate that is issued by another CA.



---

The certificate must have its `BasicConstraints` field with CA key set to `true`, and its `KeyUsage` extension field set with `keyCertSign` and/or `cRLSign`.

---

CA certificates are used to:

- Validate the certificates that remote peers present when attempting to establish a secure connection with a service on the switch, for example, the SSH server.

- Validate leaf certificates installed on the switch that are used, for example, by the syslog client, the Web UI, or REST API.

The TA profile also enables configuration of real-time checking of certificate revocation (through OCSP).

## Leaf certificates

Leaf certificates can be installed on the switch for use by features such as the syslog client, the Web UI, or REST API. If you are purchasing a certificate from a trusted CA, the switch can generate the certificate signing request (CSR) that is used to obtain the certificate. The switch can also directly generate self-signed certificates. Alternatively, the certificate and private key can be generated outside the switch and then imported. X509 certificate management software such as OpenSSL can be used to generate the private key and CSR and then combine the certificate and private key into one PEM or PKCS#12 file suitable for importation into the switch.

## Mandatory matching of peer device hostname

While validating the peer device certificates, the switch checks that the peer device configured hostname matches either the Subject Alternative Name (SAN) field or the Common Name (CN) within the certificate Subject field. If the SAN field is present and matches the hostname, validation succeeds, otherwise it fails. If the SAN field is not present, and the CN matches the hostname, validation succeeds, otherwise it fails.

## PKI EST

EST (Enrollment over Secure Transport) (RFC 7030) defines the protocol that devices use to request trusted certificate authority (CA) certificates and to enroll / re-enroll device certificates from CA services using secure channels, specifically HTTP over TLS.

Devices can be configured to request the trusted CA certificates and to request enrollment, and re-enrollment of device certificates automatically, without the need for administrator intervention, while maintaining the security and integrity of the whole enrollment process.

The switch includes an EST client implemented as a part of the PKI infrastructure.

---

For detailed CLI command descriptions, see:

- [PKI commands](#)
  - [PKI EST commands](#)
- 



## EST usage overview

- The EST client on the switch requires EST profile configuration, including EST server URL and the VRF providing HTTP connection to the EST server.
- At the time the URL is set in the EST profile, the switch connects to the EST server and downloads the trusted CA certificate chain. To accommodate CA certificate updates, the certificate chain is also downloaded before a certificate enrollment or re-enrollment is attempted.
- EST supports up to:
  - 16 EST profiles
  - 63 trusted CA certificates downloaded from EST servers.
  - 18 device certificates enrolled through EST services.
- EST profile configuration is supported through the CLI and the REST API `PKI_EST_Profile`.

- CA certificate request and device certificate enrollment is supported through the CLI and the REST custom API `CertificateManager /certificate`.

## Prerequisites for using EST for certificate enrollment

- Establish the PKI infrastructure for your organization, with the CA chain and service ready to issue certificates. Issue a service certificate for the EST server.
- Install the root CA certificate in a TA profile on the switch that will validate the EST server certificate using CLI commands `crypto pki ta-profile` and `ta-certificate`.
- Optionally, preconfigure an EST client certificate on the switch.
- Make the EST server reachable from the switch. Connect the CA service(s) to the EST server. If there is a client certificate for the EST client, install the root CA certificate on the server that will validate the client certificate.

## EST profile configuration

In the global configuration context, create an EST profile and enter its context:

```
crypto pki est-profile <EST-NAME>
```

In an EST profile context, configure the EST profile parameters using these commands:

```
url <URL>
vrf <VRF-NAME>
username <USERNAME> password [ciphertext <CIPHERTEXT-PASSWORD> |
                               plaintext <PLAINTEXT-PASSWORD>]
retry-interval <INTERVAL>
retry-count <RETRIES>
arbitrary-label <LABEL>
arbitrary-label-enrollment <LABEL>
arbitrary-label-reenrollment <LABEL>
reenrollment-lead-time <LEAD-TIME>
```

## Certificate enrollment

In the global configuration context, create a certificate and enter its context:

```
crypto pki certificate <CERT-NAME>
```

In a certificate configuration context, configure the certificate parameters:

```
key-type {rsa [key-size <K-SIZE>] | ecdsa [curve-size <C-SIZE>]}
subject [common-name <COMMON-NAME>]
        [country <COUNTRY>]
        [locality <LOCALITY>]
        [org <ORG-NAME>]
        [org-unit <ORG-UNIT>]
        [state <STATE>]
```

In a certificate configuration context, enroll the certificate using an EST service:

```
enroll est-profile <EST-NAME>
```

## Certificate re-enrollment

- The re-enrollment request is sent automatically to the same EST server that was used for the original enrollment.
- The switch presents the enrolled certificate being re-enrolled to the EST server for authentication. If the certificate has expired or authentication fails for any reason, the switch falls back to using the EST client

certificate or the username and password in the EST profile, whichever is configured, and performs a new certificate enrollment.

- Re-enrollment lead-time is configurable in the EST profile using CLI command `reenrollment-lead-time`. It sets the number of days before certificate expiry date that certificate re-enrollment will be initiated.

## Checking EST profile and certificate configuration

Show the list of EST profiles or details of a specific EST profile:

```
show crypto pki est-profile [<EST-NAME>]
```

Show a list of TA profiles whether directly configured or EST-enrolled, or details of a specific TA profile:

```
show crypto pki ta-profile [<TA-NAME>]
```

Show the list of certificates whether directly configured or EST-enrolled, or details of a specific certificate:

```
show crypto pki certificate [<CERT-NAME> [plaintext | pem]]
```

Show all certificates assigned to the switch EST client as well as certificates that are assigned to other applications on the switch.:

```
show crypto pki application
```

## EST best practices

Ensure the following:

- A time synchronization service is used on both the switch (the EST client) and the EST server.
- In all CA certificates, the `Basic Constraints` field has `CA` set to `true`, `pathlen` is set appropriately, and `Key Usage` is set with `keyCertSign`.
- In all leaf certificates, the `Extended Key Usage` field is set with the appropriate purpose as follows:
  - For server certificates, set with `serverAuth`. The `Key Usage` field has at least one of `digitalSignature`, `keyEncipherment`, or `keyAgreement`.
  - For client certificates, set with `clientAuth`. The `Key Usage` field has at least one of `digitalSignature`, or `keyAgreement`.
- The EST server is configured to include the intermediate issuer CA certificates in the trusted CA certificate chain that the EST server sends to the switch (the EST client) upon request.

## Example using EST for certificate enrollment

This example illustrates the configuration of an EST profile and enrolling application certificates using an EST server.

Prerequisites:

- An EST server is reachable from the switch management port.
- Availability of the root CA certificate used to validate the server certificate.

This example shows the following:

- Installing the root CA certificate as a TA profile for validation of the EST server certificate.
- Configuring an EST profile with the EST server information, including the username and password for client authentication and the EST server URL.
- Issuing a request to enroll a leaf certificate using the EST server.
- Assigning the enrolled certificate to the EST client and syslog client on the switch.

Each section in the below example is preceded by descriptive text.

## Example

=====

**The switch in its default configuration state.**

=====

```
switch# show running-config
Current configuration:
!
!Version ArubaOS-CX FL.10.06.0001CM
!export-password: default
user admin group administrators password ciphertext AQBapTLgcT+DNrtd0bmdXIP2L0AY
NUpwwyQEIZX4oMKtwlXcYgAAAOMKlfxH+ugf3Fe2JuWar2uKG7A/R6bqMO/ZHS364NOmpXV/Ko37ZhCq
cFpaOJsk01+IJPrukbpigCeEObM67Od8/vrASJaO6EAj+RBnWCrifwdChcUUS3XpbCUl7dmxYHNg
!
!
ssh server vrf default
ssh server vrf mgmt
vsf member 1
    type jl668a
vlan 1
spanning-tree
interface mgmt
    no shutdown
    ip dhcp
interface 1/1/1
    no shutdown
    no routing
    vlan access 1
interface 1/1/2
    no shutdown
    no routing
    vlan access 1
interface 1/1/3
    no shutdown
    no routing
    vlan access 1
...
interface 1/1/26
    no shutdown
    no routing
    vlan access 1
interface 1/1/27
    no shutdown
    no routing
    vlan access 1
interface 1/1/28
    no shutdown
    no routing
    vlan access 1
interface vlan 1
    ip dhcp
!
!
https-server vrf default
https-server vrf mgmt
switch#
```

=====

**The mgmt port is connected to a network with DNS available and the EST server reachable.**

=====

```
switch# show interface mgmt
```

```
Address Mode           : dhcp
Admin State            : up
Mac Address             : 38:21:c7:59:cd:81
IPv4 address/subnet-mask : 999.100.205.146/24
Default gateway IPv4    : 999.100.205.1
IPv6 address/prefix     :
IPv6 link local address/prefix: fe80::3a21:c7ff:fe59:cd81/64
Default gateway IPv6    :
Primary Nameserver      :
Secondary Nameserver    :
```

```
switch#
```

```
=====
Configure the root CA cert as a TA profile that will validate the server cert.
=====
```

```
switch# config
```

```
switch(config)#
```

```
switch(config)# crypto pki ta-profile root-ca-for-est-server
```

```
switch(config-ta-root-ca-for-est-server)#
```

```
switch(config-ta-root-ca-for-est-server)# ta-certificate import terminal
```

```
Paste the certificate in PEM format below, then hit enter and ctrl-D:
```

```
switch(config-ta-cert)# -----BEGIN CERTIFICATE-----
```

```
NVBAYTonfig-ta-cert)# MIIB2DCCAX6gAwIBAgIJAKtmJvZZy9RdMAoGCCqGSM49BAMCMGIXCzAJBg
QKEwNIonfig-ta-cert)# AlVTMQswCQYDVQQIEWJDQTESMBAGA1UEBxMJU9zZXZpbGx1MQwwCgYDVQ
0yMDAlonfig-ta-cert)# UEUxDjAMBgNVBAsTBUFydWJhMRQwEgYDVQQDEWtkYW5lc3Qtcml9vdDaeFw
...
```

```
YDVR0Ponfig-ta-cert)# VCnKtlhxfmV72nfxYpI979UsopuP5nCjHTAbMAwGA1UdEwQFMAMBAf8wCw
eo6yN0onfig-ta-cert)# BAQDAgEGMAoGCCqGSM49BAMCA0gAMEUCIQDb/uHvU8DFRTyfnP9wkli6sd
c=00 (config-ta-cert)# UvUO5t7/rrVxRQIgMHGjHhaNlnkjYBG8Ei3C1UDILiKlO7McMTCWVo4Ik5
```

```
switch(config-ta-cert)# -----END CERTIFICATE-----
```

```
switch(config-ta-cert)#
```

```
The certificate you are importing has the following attributes:
```

```
Subject: C = US, ST = CA, L = Roseville, O = HPE, OU = Aruba, CN = danest-root
```

```
Issuer: C = US, ST = CA, L = Roseville, O = HPE, OU = Aruba, CN = danest-root
```

```
Serial Number: 0xAB6626FXXXXD45D
```

```
TA certificate import is allowed only once for a TA profile
```

```
Do you want to accept this certificate (y/n)? y
```

```
switch(config-ta-root-ca-for-est-server)#
```

```
switch(config-ta-root-ca-for-est-server)# exit
```

```
switch(config)#
```

```
switch(config)# show crypto pki ta-profile
```

TA Profile Name	TA Certificate	Revocation Check
root-ca-for-est-server	Installed, valid	disabled

```
switch(config)#
```

```
=====
Configure the EST profile with the EST server URL, username/password.
=====
```

```
switch(config)# crypto pki est-profile test-est-server
```

```
switch(config-est-test-est-server)#
```

```
switch(config-est-test-est-server)# user fred password plaintext barney
```

```
switch(config-est-test-est-server)#
```

```
switch(config-est-test-est-server)# url https://999.0.10.229:8443/.well-known/est
```

```
switch(config-est-test-est-server)#
switch(config-est-test-est-server)# exit
switch(config)#
```

=====

**At the time the EST URL is set, the switch sends a request to the EST server to get the set of trusted CA certs. If that is successful, TA profiles will be auto-created for those CA certs.**

**Display the list of TA profiles and EST profile details.**

=====

```
switch(config)# show crypto pki ta-profile
```

TA Profile Name	TA Certificate	Revocation Check
test-est-server-est-ta00	Installed, valid	OCSP
test-est-server-est-ta02	Installed, valid	OCSP
test-est-server-est-ta05	Installed, valid	OCSP
test-est-server-est-ta01	Installed, valid	OCSP
root-ca-for-est-server	Installed, valid	disabled
test-est-server-est-ta04	Installed, valid	OCSP
test-est-server-est-ta03	Installed, valid	OCSP

```
switch(config)# show crypto pki est-profile
```

Profile Name	Downloaded TA Profiles	Enrolled Certificates
test-est-server	6	1

```
switch(config)# show crypto pki est-profile test-est-server
```

```
Profile Name      : test-est-server
Service VRF      : mgmt
Service URL       : https://999.0.10.229:8443/.well-known/est
Arbitrary Label   : not configured
Arbitrary Label Enrollment : not configured
Arbitrary Label Reenrollment : not configured
Authentication Username : fred
Authentication Password :
  AQBapR7ndgoxkMlWQUQvK+Dvd3S6m+s9fdaPQwdkMbIYEMnMBgAAAHRhliYwA==
Retry Interval    : 30 seconds
Retry Count       : 3 times
Reenrollment Lead Time : 2 days
Downloaded TA Profiles : 6
Enrolled Certificates :
  cert-for-app
switch(config)#
```

=====

**Originally, the switch only has two built-in certificates.**

=====

```
switch(config)# show crypto pki certificate
```

Certificate Name	Cert Status	EST Status	Associated Applications
local-cert	installed	n/a	captive-portal, est-client,



```

device-identity          installed          n/a          https-server, radsec-client,
                        syslog-client
                        none

switch(config)#

```

```

=====
Create a new certificate, configure its key type, key size, and subject fields.
=====

```

```

switch(config)# crypto pki certificate cert-for-app
switch(config-cert-cert-for-app)#
switch(config-cert-cert-for-app)# key-type ecdsa curve-size 521
switch(config-cert-cert-for-app)#
switch(config-cert-cert-for-app)# subject
Do you want to use the switch serial number as the common name (y/n)? n
Common Name: 999.100.205.146
Org Unit: Aruba-Roseville
Org Name: HPE
Locality: Roseville
State: CA
Country: US
switch(config-cert-cert-for-app)#

```

```

=====
Request to enroll the certificate through the EST server.
=====

```

```

switch(config-cert-cert-for-app)# enroll est-profile test-est-server
You are enrolling a certificate with the following attributes:
Subject: C=US, ST=CA, L=Roseville, OU=Aruba-Roseville, O=HPE,
        CN=999.100.205.146
Key Type: ECDSA (521)

Continue (y/n)? y
Certificate enrollment via test-est-server has been initiated. Please use
'show crypto pki certificate cert-for-app' to check its status.
switch(config-cert-cert-for-app)#

```

```

=====
Check the cert status to see if enrollment is successful. It is.
=====

```

```

switch(config-cert-cert-for-app)# show crypto pki certificate

```

Certificate Name	Cert Status	EST Status	Associated Applications
local-cert	installed	n/a	captive-portal, est-client, https-server, radsec-client, syslog-client
device-identity	installed	n/a	none
cert-for-app	installed	enroll success	none

```

switch(config-cert-cert-for-app)#
switch(config-cert-cert-for-app)# exit
switch(config)#

```

```

switch(config)# show crypto pki certificate cert-for-app pem
Certificate Name: cert-for-app
Associated Applications:
    est-client
Certificate Status: installed
EST Status: enroll success
Certificate Type: regular
Intermediates:
    Subject: C = US, ST = CA, O = HPE, OU = Aruba, CN = danest-int2
    Issuer: C = US, ST = CA, O = HPE, OU = Aruba, CN = danest-int1
    Serial Number: 0x02
    Subject: C = US, ST = CA, O = HPE, OU = Aruba, CN = danest-int1
    Issuer: C = US, ST = CA, L = Roseville, O = HPE, OU = Aruba, CN = danest-root
    Serial Number: 0x01
    Subject: C = US, ST = CA, L = Roseville, O = HPE, OU = Aruba, CN = danest-root
    Issuer: C = US, ST = CA, L = Roseville, O = HPE, OU = Aruba, CN = danest-root
    Serial Number: 0xAB6626FXXXXD45D
-----BEGIN CERTIFICATE-----
MIICizCCAjKgAwIBAgICAIGwCQYHKoZiZjOEATBOMQswCQYDVQGEwJVUzELMAkG
A1UECBMCQ0ExDDAKBgNVBAoTA0hQRTEOMAwGA1UECXMFAQXJ1YmExFDASBgNVBAMT
C2RhbmVzdC1pbmQyMB4XDTIwMTAyODE5NTczOVh0XDTIwMTEyNTE5NTczOVowbzEL
...
RTEOMAwGA1UECXMFAQXJ1YmExFDASBgNVBAMTC2RhbmVzdC1pbmQxggECMAkGBYqG
SM49BAEDSAAwRQIgVC1kVIEwXhpBSQVqVsQ36MbZrhR4XsaGbQeu7+08gbUCIQCH
cS17gcLbNxJ1WVr2jnZpPBxy9vID38FjirJiGZ5cZw==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIBpzCCAU2gAwIBAgIBAJBgcqhkJOPQQBMExCZAJBgNVBAYTA1VTMQswCQYD
VQQIEwJDQTEEMMAoGALUEChMDSFBFMQ4wDAYDVQQLEwVBcnViYTEUMBIGALUEAxML
ZGFuZXR0LWludDEwHhcNMjAwNTIwMDUyNDE5NTczOVh0XDTIwMTEyNTE5NTczOVowbzEL
...
7ovbXodgN8lqDvB11VTJY1LBSz19FKMdMBswDAYDVR0TBAUwAwEB/zALBgNVHQ8E
BAMCAQYwCQYHKoZiZjOEANJADBGAIeA+i3x7KEZsXObVruM1kwqWe+QXiLKbgNL
fL077jsSMhYCIQD/dFBkH/yN0NFzb3wi70aoo083HY2p/47t2pIBk/JNfg==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIBuTCCAWGgAwIBAgIBATAJBgcqhkJOPQQBMGIxChZAJBgNVBAYTA1VTMQswCQYD
VQQIEwJDQTESMBAGALUEBxMJUm9zZXZpbGx1MQwwCgYDVQQKEwNIUEUxDjAMBGNV
BAsTBUFydWJhMRQwEgYDVQQDEWtkYW5lc3QtcmludDEwHhcNMjAwNTIwMDUyNDE5NTczOVh0XDTIwMTEyNTE5NTczOVowbzEL
...
BgNVHRMEBTADAQH/MASGA1UdDwQEAwIBBjAJBgcqhkJOPQQBA0cAMEQCIGr1ZmBX
SmbhDvG9pRiXG0YmQVbvZd37jRQdE+mEk2jfAiBFghZmJUadhQbuPUTNs9A7bdYk
wej0mJe5bRpd7sqwRQ==
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIB2DCCAX6gAwIBAgIJAKtmJvZZy9RdMAoGCCqGSM49BAMCMGIxChZAJBgNVBAYT
A1VTMQswCQYDVQQIEwJDQTESMBAGALUEBxMJUm9zZXZpbGx1MQwwCgYDVQQKEwNI
UEUxDjAMBGNVBAsTBUFydWJhMRQwEgYDVQQDEWtkYW5lc3QtcmludDEwHhcNMjAwNTIwMDUyNDE5NTczOVh0XDTIwMTEyNTE5NTczOVowbzEL
...
VCnKTLhxfmV72nfXypI979UsopuP5nCjHTAbMAwGA1UdEwQFMAMBAf8wCwYDVROp
BAQDAgEGMAoGCCqGSM49BAMCA0gAMEUCIQDb/uHvU8DFRTyfnP9wk1i6sdeo6yN0
UvUO5t7/rrVxRQIgMHGjHhaN1nkjYBG8Ei3C1UDILiK107McMTCWVo4Ik5c=
-----END CERTIFICATE-----
switch(config)#

```

```

=====
Initially, all applications use the default local-cert.
=====

```

```

switch(config)# show crypto pki application

```

Associated Applications	Certificate Name	Cert Status
captive-portal		not configured, using local-cert
est-client		not configured, using local-cert
https-server		not configured, using local-cert
radsec-client		not configured, using local-cert
syslog-client		not configured, using local-cert
switch(config)#		

=====  
**Assign the newly enrolled cert to applications as desired.**  
**In this example, the cert is assigned to the est-client and syslog.**  
 =====

```
switch(config)# crypto pki application est-client certificate cert-for-app
switch(config)# crypto pki application syslog-client certificate cert-for-app
switch(config)#
switch(config)# show crypto pki application
```

Associated Applications	Certificate Name	Cert Status
captive-portal		not configured, using local-cert
est-client	cert-for-app	valid
https-server		not configured, using local-cert
radsec-client		not configured, using local-cert
syslog-client	cert-for-app	valid
switch(config)#		

## Example including the use of an intermediate certificate

This example shows the following:

- Installing a root CA as a TA profile.
- Creating a CSR for a leaf certificate.
- Installing the signed leaf certificate issued by an intermediate CA. The intermediate CA certificate is included after the signed leaf certificate.

Each section in the below example is preceded by descriptive text.

### Example

=====  
**Install root CA as a TA profile**  
 =====

```
switch(config)# crypto pki ta-profile root
switch(config-ta-root)# ta-certificate import terminal
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-ta-cert)# -----BEGIN CERTIFICATE-----
switch(config-ta-cert)# MIIGATCCA+mgAwIBAgIJAL/JIZfJ0GpcMA0GCSqGSIUAMIGOMQswCQYD
switch(config-ta-cert)# VQQGEwJVUzETMBEGA1UECAwKQ2FsaWZvcn5pYTESBwwJUm9zZXZpbGx1
switch(config-ta-cert)# MQwwCgYDVQQKDANIUEUxEzARBgNVBASMcK5ldmcxFTATBgNVBAMMDFR1
...
switch(config-ta-cert)# rvadRXSAsUlevJRNNOyINrEJyOfUX2hAfLaiBYP+In6gKTAWVh1xLiXn
switch(config-ta-cert)# LlryAb2/go4BTYjil3eJyXxweUHheuBeesEslBawLv0cPCQPTTdbc970
switch(config-ta-cert)# iWbyAmfSpD/TS3AgCLnBFPKEKsmsOf0LF3/C9dRUXjIHT/LDBr+lgzY3
```

```

switch(config-ta-cert)# m2NCvxY=
switch(config-ta-cert)# -----END CERTIFICATE-----
switch(config-ta-cert)#
The certificate you are importing has the following attributes:
Subject: C = US, ST = California, L = Roseville, O = HPE, OU = Networking,
        CN = Test CA root, emailAddress = generic@corp.com
Issuer:  C = US, ST = California, L = Roseville, O = HPE, OU = Networking,
        CN =Test CA root, emailAddress = generic@corp.com
Serial Number: 0xBFC92197xxxxxxxx
TA certificate import is allowed only once for a TA profile
Do you want to accept this certificate (y/n)? y
switch(config-ta-root)# exit

```

```

=====
Create a CSR for a leaf certificate
=====

```

```

switch(config)# crypto pki certificate leaf
switch(config-cert-leaf)# subject
Do you want to use the switch serial number as the common name (y/n)? y
Common Name: SG9Zxxxxxx
Org Unit:
Org Name:
Locality:
State:
Country:
switch(config-cert-leaf)# enroll terminal
You are enrolling a certificate with the following attributes:
Subject: C=<empty>, ST=<empty>, L=<empty>, OU=<empty>, O=<empty>,
        CN=SG9Zxxxxxx
Key Type: RSA (2048)

Continue (y/n)? y

```

```

-----BEGIN CERTIFICATE REQUEST-----
MIICWjCCAUICAQIwFTETMBEGA1UEAwwKU0c5WktONDAwSoZlhvcN
AQEBBQADggEPADCCAQoCggEBAMKdtoucDEMeuZjPGvCcWTm4D39A
WBA8K/bduJvM1E2B/uirU2TX7mF6lN30akClSxZOoofZAmBPCzI3
...
wZtb5c8fYCSR+TpLwZAdoXrvGJqJ1aGzV6/kVfb7rM6ulBksfBo/
JwO+7x8Vn5h1dGCrsl9CPJienni/fq24+1CJzspMbY9BKu9EIL+P
5ND9BmN0IzEmDO26F+Ip74DqFCIYjXt13uPJk4cwJkXq121hlcrG
UlatpvjNEpZotfoEryDJSs0pHXky7VjltYABiuDy
-----END CERTIFICATE REQUEST-----

```

```

=====
Install the signed leaf certificate issued by an intermediate CA. The
lintermediate CA certificate is included after the signed leaf certificate.
=====

```

```

switch(config-cert-leaf)# import terminal ta-profile root
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-cert-import)# -----BEGIN CERTIFICATE-----
switch(config-cert-import)# MIEKTCCAhGgAwIBAgIJA01LS0BmKxtbMA0GCSqGSIYxCzAJBgNV
switch(config-cert-import)# BAYTAKFVMRUwEwYDVQQIDAxJbnRlcm1lZG9VBAoMGE1udGVybmV0
switch(config-cert-import)# IFdpZGdpdHMgUHR5IEExOZDENMAsGA1UEAw0yMDA1MTQyMDI3MTla
...
switch(config-cert-import)# axnZcIaNp4eNi95in+TvckXA0eMLScNyR7IF+Wjn56H0fQKYsHp/
switch(config-cert-import)# jllbCkyBlxKnn6IpzIj/hvAx3NpA0jXx/qJA+V/cltaAL6+QPZmI
switch(config-cert-import)# vr5GZsoV72BHFOXxoteZlmWMUdVldYXXP2DzEUbttr9zojwz0MyK

```

```

switch(config-cert-import)# Qz5tc0BlGfJAtghykw==
switch(config-cert-import)# -----END CERTIFICATE-----
switch(config-cert-import)# -----BEGIN CERTIFICATE-----
switch(config-cert-import)# MIIFyzCCA7OgAwIBAgIJA01LSoBmKxtwMA0GCSqGCIgOMQswCQYD
switch(config-cert-import)# VQQGEwJVUzETMBEGA1UECAwKQ2FsaWZvc1UEBwwJUm9zZXZpbGxl
switch(config-cert-import)# MQwwCgYDVQQKDANIUEUxEzARBgNVBAsMCmcxFTATBgNVBAMMDFRl
...
switch(config-cert-import)# LM9DV3YNWOM4UMMP2HXaDDfqxZPX9Zsj6Gl/stRCh8SVfsF2duYR
switch(config-cert-import)# 5brLfEpiDhXrZVXxF91ljRAB02JPLSUufg7xr6M/K5aCujxVYzK7
switch(config-cert-import)# DQaCEw5NlmC1vpYlY2TG3dlUQPZDeQOAHwuBd4HewqDHWfp/T04=
switch(config-cert-import)# -----END CERTIFICATE-----
switch(config-cert-import)#
Leaf certificate is validated with root and imported successfully.
switch(config-cert-leaf)#

```

## Installing a self-signed leaf certificate (created inside the switch)

This procedure describes how to create (wholly inside the switch) and install a self-signed X.509 leaf certificate. And associate it with one of the following switch features: syslog client, RadSec client, captive-portal, HTTPS server, or HSC (hardware switch controller).

### Procedure

1. Create a leaf certificate context with the command `crypto pki certificate`. This switches to the leaf certificate configuration context.
2. Define leaf certificate properties with the command `subject`.
3. Set the encryption key type for the leaf certificate with the command `key-type`.
4. Generate and install the self-signed certificate with the command `enroll self-signed`.
5. Exit the leaf certificate context with the command `exit`.
6. Associate the leaf certificate with a switch feature (syslog client, RadSec client, captive-portal, HTTPS server, or HSC) with the command `crypto pki application`.

### Example

This example:

- Creates the leaf certificate context.
- Defines the leaf certificate characteristics.
- Creates and installs the self-signed leaf certificate.
- Associates the leaf certificate with the syslog client (application) on the switch.

```

switch(config)# crypto pki cert SS_LC
8400X(config-cert-SS_LC)# subject common-name SSLeaf country US
state CA locality Rocklin org Company org-unit Site
8400X(config-cert-SS_LC)# key-type rsa key-size 3072
8400X(config-cert-SS_LC)# enroll self-signed
You are enrolling a certificate with the following attributes:
Subject: C=US, ST=CA, L=Rocklin, OU=Site, O=Company,
        CN=SSLeaf
Key Type: RSA (3072)

Continue (y/n)? y

```

```
Self-signed certificate is created and enrolled successfully.
8400X(config-cert-SS_LC)# exit
switch(config)# crypto pki application syslog-client certificate SS_LC
```

## Installing a self-signed leaf certificate (created outside the switch)

This procedure describes how to install a self-signed X.509 leaf certificate (that was created outside the switch). And then associate the certificate with one of the following switch features: syslog client, RadSec client, captive-portal, HTTPS server, or HSC (hardware switch controller).

### Prerequisites

A self-signed leaf certificate (including private-key data) must be created outside the switch.

### Procedure

1. Create the leaf certificate context with the command `crypto pki certificate` which then switches to the created leaf certificate context.
2. Import the leaf certificate data into the switch with the command `import (self-signed leaf certificate)`.
3. Exit the leaf certificate context with the command `exit`.
4. Associate the leaf certificate with a switch feature (syslog client, RadSec client, captive-portal, HTTPS server, or HSC) with the command `crypto pki application`.

### Example

This example:

- Creates the leaf certificate context.
- Imports the self-signed leaf certificate.
- Associates the leaf certificate with the syslog client (application) on the switch.

```
switch(config)# switch(config)# crypto pki certificate SS_LC2
switch(config)# switch(config-cert-SS_LC)# import terminal self-signed
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-cert-import)# -----BEGIN CERTIFICATE-----
switch(config-cert-import)# MIIFRDCCAyygAwIBAgIQP8nnS2Vp15u07xXMdktdJzANBgkqhkiG9
switch(config-cert-import)# MQswCQYDVQGEwJVUEOMAwGA1UECgwFXJlYmxDAOgNBAMMB1Jvb3gw
switch(config-cert-import)# HhcNMTkNDEwMjIwNTIWhcJiIwMTA0MjIwNE1WjBzQswQYDVQGEwJV
...
switch(config-cert-import)# 1fIYZYQYla0AwFuPTTxBXHYwRxTPbUYU5tumJrfwRpmE4OVY8S9D
switch(config-cert-import)# 1NGNm3NG03GqPScs/TF9bVyFA5BOrS5lmm7kNfRYlK8D/kMTfRreS
switch(config-cert-import)# YQ1ulNqShps=
switch(config-cert-import)# -----END CERTIFICATE-----
switch(config-cert-import)# -----BEGIN ENCRYPTED PRIVATE KEY-----
switch(config-cert-import)# MIIFDjBABgkqhkiG9wBBQ0wMzAbBgkqhkiG9wQwDQImNpJMN7sVGwC
switch(config-cert-import)# MBQGCCqGSib3DQMHAit+2qadNAASCMg5LYJ4AFm3EffhH5p51Ggr8
switch(config-cert-import)# IJ6L/UhEtH523nUkdV6gvoAWgoYaeD83PeswToAGv5VS8OMFTPttr
...
switch(config-cert-import)# OgSecqZsG6arbx0ESaYBir1c/6rPs1pcjbDxw283DiD1MW0peoS2a
switch(config-cert-import)# iKnXnUMpVPfLc74ty2S41DtH0X9Sgf6aalLjiStg+N7cND9XfGtj/
switch(config-cert-import)# cb4=
```

```
switch(config-cert-import)# -----END ENCRYPTED PRIVATE KEY-----
switch(config-cert-import)#
Enter import password: *****
Leaf certificate is validated as self-signed certificate and imported successfully.
switch(config-cert-SS_LC2)# exit
switch(config)# crypto pki application syslog-client certificate SS_LC2
```

## Installing a certificate of a root CA

### Prerequisites

- A certificate of a root CA (that is used as the signer).
- Revocation checking URLs for the CA (optional).

### Procedure

1. Create a TA profile with the command `crypto pki ta-profile` which then switches to the created TA profile context.




---

Step 2 is optional and suggested only for advanced users.

---

2. Optionally enable certificate revocation checking with the command `revocation-check ocs`. Most certificates contain revocation checking URLs for OCS. If you want to override these URLs, configure custom revocation checking URLs with the command `ocsp url`.
3. Import the certificate of the root CA with the command `ta-certificate`.

### Example

This example installs the certificate **root-cert** and defines custom revocation checking URLs:

```
switch(config)# crypto pki ta-profile root-cert
switch(config-ta-root-cert)# revocation-check ocs
switch(config-ta-root-cert)# ocsp url primary http://ocsp-server.site.com
switch(config-ta-root-cert)# ocsp url secondary http://ocsp-server2.site.com
switch(config-ta-root-cert)# ta-certificate import terminal
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-ta-cert)# -----BEGIN CERTIFICATE-----
switch(config-ta-cert)# MIIDuTCCAqECCQCuoXeJ2ZNYcjANBgkqhkiG9w0BAQsFADCBQzELMAEBh
switch(config-ta-cert)# VVMxEzARBgNVBAgMCKNhGlmb3JuaWExEDAObgNVBACMB1JvY2tsDAKBg
switch(config-ta-cert)# BAOMA0hQTjEVMBMGA1UECwwMSFBOUm9zZXZpbGx1MSowKAYDVQOCG5zd
...
switch(config-ta-cert)# x3Wff3dFZ8o9sd5LVAHneH/ztb9MP34z+le1V346r12L2kpxmTOVJVyTO
switch(config-ta-cert)# BIzD/ST/HaWI+OS+S80rm93PSscEbb9GWk7vshh5EnW/moehBKcE40lzy
switch(config-ta-cert)# 3LvMLZcssSe5J2Ca2XIhfDme8UaNZ7syGYMSAW0nG7yYHWkEOQu9s
switch(config-ta-cert)# -----END CERTIFICATE-----
switch(config-ta-cert)#
The certificate you are importing has the following attributes:
Issuer: C=US, ST=CA, L=Rocklin, O=Company, OU=Site,
       CN=site.com/emailAddress=test.ca@site.com
Subject: C=US, ST=CA, L=Rocklin, O=Company, OU=Site,
        CN=8400/emailAddress=test.ca@site.com
Serial Number: 12121221634631568498 (0xaea51217d5945772)

TA certificate import is allowed only once for a TA profile
```

```
Do you want to accept this certificate (y/n)? y
TA certificate accepted.
switch(config-ta-root-cert) #
```

## Installing a CA-signed leaf certificate (initiated in the switch)

This procedure describes how to create and install an X.509 leaf certificate that is initiated inside the switch but signed outside the switch by a CA. And then associate the certificate with one of the following switch features: syslog client, RadSec client, HTTPS server, or HSC (hardware switch controller).

### Prerequisites

Root CA certificate `root-cert` must be installed as described in [Installing a certificate of a root CA](#).

### Procedure

1. Create a leaf certificate context with the command `crypto pki certificate` which then switches to the created leaf certificate configuration context.
2. Define leaf certificate properties with the command `subject`.
3. Set the encryption key type for the leaf certificate with the command `key-type`.
4. Generate the certificate signing request (CSR) with the command `enroll terminal`.
5. Use the CSR to obtain a leaf certificate from the root CA, using the root CA directly as the signer CA.
6. Import the leaf certificate into the switch with the command `import (CA-signed leaf certificate)`.
7. Exit the leaf certificate context with the command `exit`.
8. Associate the leaf certificate with a switch feature (syslog client, RadSec client, HTTPS server, or HSC) with the command `crypto pki application`.

### Example

This example:

- Creates the leaf certificate context.
- Defines the leaf certificate characteristics.
- Generates the leaf certificate signing request in the switch for getting signed outside the switch by a CA.
- Imports the CA-signed leaf certificate into the switch.
- Associates the leaf certificate with the syslog client (application) on the switch.

```
switch(config)# crypto pki certificate lcert
switch(config-cert-lcert)# subject common-name Leaf country US state CA
locality Rocklin org Company org-unit Site
switch(config-cert-lcert)# key-type rsa key-size 3072
switch(config-cert-lcert)# enroll terminal
You are enrolling a certificate with the following attributes:
Subject: C=US, ST=CA, L=Rocklin, O=Company, OU=Site
        CN=Leaf
Key Type: RSA (2048)

Continue (y/n)? y
```



```

-----BEGIN CERTIFICATE REQUEST-----
MIIBozCCAQwCAQAwYzEVMBMGAlUEAxMMcG9kMDEtODQwMC0xMQ4wDAYDV
nViYTEMMAoGAlUEChMDSFBFMRIwEAYDVQQHEw1Sb3Nldm1sbGUxCzAJBg
NBMQswCQYDVQQGEwJVUzCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYE
...
GBAJ4L3lFFfWBEL+KAKpOGjZcVmw1BMqSKFtOFNF9nzmUmONmU3SKy6dz
7Au22mf3lWDxzrtCC/dj5RtWJeJekxp2LCIK/3eRXUwbYveQDKcxH7j9Z
ace+2tA68F2vlgRCQ/hcQH0YmNuaq4Ne3w0dhm7HlUrx
-----END CERTIFICATE REQUEST-----

switch(config-cert-lcert)# import terminal ta-profile root-cert
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-cert-import)# -----BEGIN CERTIFICATE-----
switch(config-cert-import)# MIIFRDCCAYygwIBAgIQPnnS2Vp5u07XMdktDJzANBgkqhkiG9w0Bv
switch(config-cert-import)# MQswCQYDVQQGEwJVEOMAwG1UECgwFJlYmxDAOgNBMMB1Jvb3QgQ0Ew
switch(config-cert-import)# HhcNMTkNDEwMjIwNTWcjIwMTA0MjwNE1WBzQswQYDVQQGEwJVUzEL
...
switch(config-cert-import)# 1fIYZYGQyla0AwFuTtXBXyWRxPbUYU5tumrfwRPmE4OVY8S9DQgcr
switch(config-cert-import)# lNGNm3NG03GqPcs/T9bVyF5BOrS5lmm7kNfRYl8D/kMTfRreSdxis
switch(config-cert-import)# YQ1ulNqShps=
switch(config-cert-import)# -----END CERTIFICATE-----
switch(config-cert-import)#
Leaf certificate is validated with root-cert and imported successfully.
switch(config-cert-lcert)# exit
switch(config)# crypto pki application syslog-client certificate lcert

```

## Installing a CA-signed leaf certificate (created outside the switch)

This procedure describes how to install an X.509 leaf certificate that was created and signed (by a CA) outside the switch. And then associate the certificate with one of the following switch features: syslog client, RadSec client, captive-portal, HTTPS server, or HSC (hardware switch controller).

### Prerequisites

- Root CA certificate `root-cert` installed as described in [Installing a certificate of a root CA](#).
- A CA-signed leaf certificate (including private-key data) created outside the switch.

### Procedure

1. Create the leaf certificate context with the command `crypto pki certificate` which then switches to the created leaf certificate context.
2. Import the leaf certificate into the switch with the command `import (CA-signed leaf certificate)`.
3. Exit the leaf certificate context with the command `exit`.
4. Associate the leaf certificate with a switch feature (syslog client, RadSec client, captive-portal, HTTPS server, or HSC) with the command `crypto pki application`.

### Example

This example:

- Creates the leaf certificate context.
- imports the CA-signed leaf certificate.
- Associates the leaf certificate with the syslog client (application) on the switch.

```
switch(config)# switch(config)# crypto pki certificate CA_LC
switch(config)# switch(config-cert-CA_LC)# import terminal ta-profile root-cert
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-cert-import)# -----BEGIN CERTIFICATE-----
switch(config-cert-import)# MIIFRDCCAyygAwIBAgIQP8nn2Vp15u07XMktDJANBgkqhkiG9w0Bv
switch(config-cert-import)# MQswCQYDVQGEwJVUEOMAw1UECgwFX1YmxDOgNBAMMB1Jvb3QgQ0Ew
switch(config-cert-import)# HhcNMTkNDEwMjIwNT1WWhjIMTAOMjIwNE1jBzQswYDVQQGEwJVUzEL
...
switch(config-cert-import)# 1fIYZYGQyla0AwFuPTTxBXHYRxTPbUYUtmJrwrPmE4OVY8S9DQgcr
switch(config-cert-import)# lNGNm3NG03GqPScs/TF9bVyFABOrlmm7kNfRlK8D/kMTfRreSdxis
switch(config-cert-import)# YQ1ul1NqShps=
switch(config-cert-import)# -----END CERTIFICATE-----
switch(config-cert-import)# -----BEGIN ENCRYPTED PRIVATE KEY-----
switch(config-cert-import)# MIIFDjBABgkqhkiG9wBBQ0wMzAbBgkiwQwwQImNpJMN7sVGwCaggA
switch(config-cert-import)# MBQGCCqGSib3DQMHAit+2qadNAASCMgLYJ4AFEfhH5p51Ggr86VqS
switch(config-cert-import)# IJ6L/UhEtH523nUkdV6gvoAWgoYaeD8eswAGv5VS8OMFTPttrn5/K
...
switch(config-cert-import)# OgSecqZsG6arbx0ESaYBirlc6rPslpcbDx283DD1MWOpes2aEmOX
switch(config-cert-import)# iKnXnUMpVPfLc74ty2S41tH0X9gfaa1LiStg+N7cND9XfGtjaV2+/
switch(config-cert-import)# cb4=
switch(config-cert-import)# -----END ENCRYPTED PRIVATE KEY-----
switch(config-cert-import)#
Enter import password: *****
Leaf certificate is validated with root-cert and imported successfully.
switch(config-cert-CA_LC)# exit
switch(config)# crypto pki application syslog-client certificate CA_LC
```

## PKI commands

### crypto pki application

#### Syntax

```
crypto pki application <APP-NAME> certificate <CERT-NAME>
no crypto pki application <APP-NAME> certificate
```

#### Description

Associates a leaf certificate with a feature (application) on the switch. By default, all features are associated with the default, self-signed certificate `local-cert`. This certificate is created by the switch the first time it starts.

The `no` form of this command associates the specified feature with the default certificate.

#### Command context

config

#### Parameters

<APP-NAME>

Specifies the name of a feature on the switch:

- `captive-portal`: Captive portal

- `est-client`: EST client
- `hsc`: Hardware switch controller.
- `https-server`: HTTPS server.
- `radsec-client`: RadSec client.
- `syslog-client`: Syslog client.

<CERT-NAME>

Specifies the name of an installed leaf certificate.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Associating the EST client with leaf certificate **leaf-cert1**:

```
switch(config)# crypto pki application est-client certificate leaf-cert1
```

Associating the syslog client with leaf certificate **leaf-cert**:

```
switch(config)# crypto pki application syslog-client certificate leaf-cert
```




---

`syslog-client` communicates with syslog server over TLS.

You can associate a certificate with the `syslog-client` application by enrolling the certificate manually or through EST.

---

Setting the syslog client to use the default certificate:

```
switch(config)# no crypto pki application syslog-client certificate
```

Setting the RadSec client to use the default certificate:

```
switch(config)# no crypto pki application radsec-client certificate
```

Associating the RadSec client with leaf certificate **leaf-cert**:

```
switch(config)# crypto pki application radsec-client certificate leaf-cert
```

Associating the HTTPS server with leaf certificate **leaf-cert2**:

```
switch(config)# crypto pki application https-server certificate leaf-cert2
```

## crypto pki certificate

### Syntax

`crypto pki certificate <CERT-NAME>`

```
no crypto pki certificate <CERT-NAME>
```

## Description

Creates a leaf certificate and changes to its context `config-cert-<CERT-NAME>`. If the specified leaf certificate exists, this command changes to its context.

The first time the switch starts it creates a self-signed, default leaf certificate called `local-cert`. This certificate is used by any switch application that does not have an associated leaf certificate.

The `no` form of this command deletes the specified leaf certificate. The default leaf certificate `local-cert` cannot be deleted.

## Command context

`config`

## Parameters

`<CERT-NAME>`

Specifies the name of a leaf certificate. Range: 1 to 32 alphanumeric characters (excluding ").

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Creating leaf certificate **leaf-cert**:

```
switch(config)# crypto pki certificate leaf-cert
switch(config-cert-leaf-cert)#
```

Deleting leaf certificate **leaf-cert**:

```
switch(config)# no crypto pki certificate leaf-cert
The leaf certificate has associated applications. Deleting the certificate
will make the applications use the default certificate local-cert.
Continue (y/n)? y
switch(config)#
```

# crypto pki ta-profile

## Syntax

```
crypto pki ta-profile <TA-NAME>
no crypto pki ta-profile <TA-NAME>
```

## Description

Creates a trust anchor (TA) profile and changes to the `config-ta-<TA-NAME>` context for the profile. Each TA profile stores the certificate for a trusted CA. Up to 64 profiles can be defined.

If the specified TA profile exists, this command changes to the `config-ta-<TA-NAME>` context for the profile.

The `no` form of this command removes the specified TA profile.



---

When creating a new profile, If you exit the `config-ta-<TA-NAME>` context without importing the TA certificate, the profile is discarded.

---

## Command context

config

## Parameters

<TA-NAME>

Specifies the TA profile name. Range: 1 to 48 alphanumeric characters excluding ".



The TA profile name cannot end with `est-ta<nn>` where `<nn>` is 00 to 99. For example, `company-trust-anchor-est-ta01` is not allowed. This TA profile name suffix is reserved for TA profiles that are created for CA certificates from EST servers.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Creating the TA profile **root-cert**:

```
switch(config)# crypto pki ta-profile root-cert
switch(config-ta-root-cert)#
```

Removing TA profile **root-cert**:

```
switch(config)# no crypto pki ta-profile root-cert
```

## enroll self-signed

### Syntax

`enroll self-signed`

### Description

Generates a key pair and generates a self-signed certificate with it.

The subject fields and key type of the current leaf certificate must be defined before running this command. If not, you are prompted to fill in the subject fields, and the key type is set to `RSA 2048`.

### Command context

`config-cert-<CERT-NAME>`

### Authority

Administrators or local user group members with execution rights for this command.

### Example

Enrolling the leaf certificate **leaf-cert**:

```
switch(config-cert-leaf-cert)# enroll self-signed
You are enrolling a certificate with the following attributes:
Subject: C=US, ST=CA, L=Rocklin, OU=Site, O=Comp,
        CN=Leaf01
Key Type: RSA (2048)
```

```
Continue (y/n)? y
Self-signed certificate is created and enrolled successfully.

switch(config-cert-leaf-cert)#
```

## enroll terminal

### Syntax

```
enroll terminal
```

### Description

Generates a key pair and certificate signing request (CSR) for the current leaf certificate. Use the CSR to obtain a signed certificate from a certificate authority (CA), and then import the certificate onto the switch with the command `import terminal`.

The key type, and the certificate common name in the subject fields of the current leaf certificate must be completed before running this command.

### Command context

```
config-cert-<CERT-NAME>
```

### Authority

Administrators or local user group members with execution rights for this command.

### Example

Enrolling the leaf certificate **leaf-cert**:

```
switch(config-cert-leaf-cert)# enroll terminal
You are enrolling a certificate with the following attributes:
Subject: C=US, ST=CA, L=Rocklin, OU=Site, O=Comp,
        CN=Leaf01
Key Type: RSA (2048)

Continue (y/n)? y

-----BEGIN CERTIFICATE REQUEST-----
MIIBozCCAQwCAQAwYzEVMBMGA1UEAxMMcG9kMDEtODQwMC0xMQ4wDAYDVQQLEwV
nViYTEMMAoGAlUEChMDSFBFMRIwEAYDVQQHEw1Sb3Nldm1sbGUxCzAJBgNVBAGT
NBMQswCQYDVQQGEwJVUzCBnzANBjQkqhkiG9w0BAQEFAAOBjQAwgYkCgYEAtKcLS
...
GBAJ4L3lFFfWBEL+KAKpOGjZcVmw1BMqSKFtOFNF9nzmUmONmU3SKy6dzQ+6ynR
7Au22mf3lWDxzrtCC/dj5RtWJeJekxp2LCIK/3eRXUwbYveQDKcxH7j9ZB+BAp2
ace+2tA68F2vlgRCQ/hcQH0YmNuaq4Ne3w0dhm7HlUrx
-----END CERTIFICATE REQUEST-----
switch(config-cert-leaf-cert)#
```

## import (CA-signed leaf certificate)

### Syntax

```
import terminal ta-profile <TA-NAME> [password <PW>]
import <REMOTE-URL> ta-profile <TA-NAME> [password <PW>] [vrf <VRF-NAME>]
import <STORAGE-URL> ta-profile <TA-NAME> [password <PW>]
```

### Description

Imports a CA-signed leaf certificate and then validates the certificate against the specified TA profile. If the imported data includes a private key, the private key must match the leaf certificate being imported. If the imported data does not include a private key, the certificate must match a CSR that was previously generated with the command `enroll terminal` and must be signed by the CA whose root certificate is installed in the specified TA profile. The TA profile must exist and have a TA certificate configured.

## Parameters

`terminal`

Import the certificate by pasting PEM-format data at the console. Upon execution, the `config-cert-import` context is entered for certificate pasting. To complete certificate data entry press Control-D in your terminal program. Alternatively, the pasted certificate data can include at its end the delimiter `END_OF_CERTIFICATE` (after the `-----END CERTIFICATE-----` line), making entry of Control-D unnecessary.

`ta-profile <TA-NAME>`

Specifies the TA profile name. Range: 1 to 48 alphanumeric characters excluding ".

`<REMOTE-URL>`

Specifies a certificate data file on a remote TFTP or SFTP server. The URL syntax is:

`{tftp:// | sftp://<USER>@} {<IP> | <HOST>} [:<PORT>] [;blocksize=<SIZE>]/<FILE>`

`<STORAGE-URL>`

Available on switch families that provide USB device file import capability, specifies a certificate data file on a USB storage device inserted in the switch USB port. The URL syntax is:

`usb:/<FILE>`

`password <PW>`

Specifies the plaintext password used to decrypt the private key in the imported certificate data. When this parameter is omitted, the password is prompted for as required. Range: 1 to 32 alphanumeric characters.

`vrf <VRF-NAME>`

Specifies the name of the VRF to use for the remote URL file transfer. The default is `mgmt`.

## Command context

`config-cert-<CERT-NAME>`

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

- The imported data must include all the intermediate CA certificates in the certificate chain leading to the certificate imported into the specified TA profile.
- This command cannot be used with the default certificate `local-cert`.
- The PEM data format is supported for all import sources. The PKCS#12 data format is supported for `<REMOTE-URL>` and `<STORAGE-URL>`.
- The PEM data must be delimited with these lines for the certificate data:

```
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
```

And the PEM data must be delimited with either of these line pairs for the private key data:

```
-----BEGIN PRIVATE KEY-----
-----END PRIVATE KEY-----
```

```
-----BEGIN ENCRYPTED PRIVATE KEY-----
-----END ENCRYPTED PRIVATE KEY-----
```

## Examples

Importing a leaf certificate from the console:

```
switch(config)# crypto pki certificate leaf-cert
switch(config-cert-leaf-cert1)# import terminal ta-profile root-cert
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-cert-import)# -----BEGIN CERTIFICATE-----
switch(config-cert-import)# MIIFRDCCAyqAwIBAgQFP8nS2Vp15u0xXMdkDJzANBgkqhkiG9w0Bv
switch(config-cert-import)# MQswCQYDVQGEwJVUEOMAwGA1UCgwFXJ1YmDagNBAMM1Jvb3QgQ0Ew
switch(config-cert-import)# HhcNMTkNDEwMjIwNTIWhcjIwMTOMjwNE1WjzQswQDVQGEwJVUzEL
...
switch(config-cert-import)# 1fIYZYGQyla0AwFuPTTxBXHYwRxTPbUYU5umJfRPmE4VY8S9DQgcr
switch(config-cert-import)# 1NGNm3NG03GqPScs/TF9bVyFA5BOS5lmmkfRYK8D/kMTfRreSdxis
switch(config-cert-import)# YQ1ul1NqShps=
switch(config-cert-import)# -----END CERTIFICATE-----
switch(config-cert-import)# -----BEGIN ENCRYPTED PRIVATE KEY-----
switch(config-cert-import)# MIIFDjBABGkqhkiG9wBBQ0wMzAbBgqkw0QwwDQIpJMN7sVGwCaggA
switch(config-cert-import)# MBQGCCqGS1b3DQMHAit+2qadNAASCgLYJ4Am3EfhH5p51Ggr86VqS
switch(config-cert-import)# IJ6L/UhEtH523nUkdV6gvAgoYaD83PswToAGv5VS8OMFTPttrn5/K
...
switch(config-cert-import)# OgSecqZsG6arbx0ESaYBir1c/6rPspcjbx283iD1MWOpes2aEmOX
switch(config-cert-import)# iKnXnUMpVPfLc74ty2S41DtH0X9gf6aaljStg+7cND9XfgTjaV2+/
switch(config-cert-import)# cb4=
switch(config-cert-import)# -----END ENCRYPTED PRIVATE KEY-----
switch(config-cert-import)#
Enter import password: *****
Leaf certificate is validated with root-cert and imported successfully.
switch(config-cert-leaf-cert)#
```

Importing a leaf certificate from a remote file:

```
switch(config)# crypto pki certificate leaf-cert2
switch(config-cert-leaf-cert2)# import tftp://1.1.1.2/c2.p12 ta-profile root-cert
% Total      % Received % Xferd  Average Speed   Time    Time       Time  Current
           Dload    Upload   Total     Spent    Left     Speed
100  3722  100  3722    0     0  391k      0  --:--:--  --:--:--  --:--:--  391k
100  3722  100  3722    0     0  376k      0  --:--:--  --:--:--  --:--:--  376k
Enter import password: *****
Leaf certificate is validated with root-cert and imported successfully.
switch(config-cert-leaf-cert2)#
```

## import (self-signed leaf certificate)

### Syntax

```
import terminal self-signed [password <PW>]
import <REMOTE-URL> self-signed [password <PW>] [vrf <VRF-NAME>]
import <STORAGE-URL> self-signed [password <PW>]
```

### Description

Imports a self-signed leaf certificate including its matching private key.

### Parameters

terminal

Import the certificate by pasting PEM-format data at the console. Upon execution, the `config-cert-import` context is entered for certificate pasting. To complete certificate data entry press Control-D in your terminal program. Alternatively, the pasted certificate data can include at its end the delimiter `END_OF_CERTIFICATE` (after the `-----END CERTIFICATE-----` line), making entry of Control-D unnecessary.



<REMOTE-URL>

Specifies a certificate data file on a remote TFTP or SFTP server. The URL syntax is:

```
{tftp:// | sftp://<USER>@} {<IP>|<HOST>} [:<PORT>] [;blocksize=<SIZE>]/<FILE>
<STORAGE-URL>
```

Available on switch families that provide USB device file import capability, specifies a certificate data file on a USB storage device inserted in the switch USB port. The URL syntax is:

usb:/<FILE>

password <PW>

Specifies the plaintext password used to decrypt the private key in the imported certificate data. When this parameter is omitted, the password is prompted for as required. Range: 1 to 32 alphanumeric characters.

vrf <VRF-NAME>

Specifies the name of the VRF to use for the remote URL file transfer. The default is `mgmt`.

## Command context

config-cert-<CERT-NAME>

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

- This command cannot be used with the default certificate `local-cert`.
- The PEM data format is supported for all import sources. The PKCS#12 data format is supported for <REMOTE-URL> and <STORAGE-URL>.
- The PEM data must be delimited with these lines for the certificate data:

```
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
```

And the PEM data must be delimited with either of these line pairs for the private key data:

```
-----BEGIN PRIVATE KEY-----
-----END PRIVATE KEY-----
```

```
-----BEGIN ENCRYPTED PRIVATE KEY-----
-----END ENCRYPTED PRIVATE KEY-----
```

## Example

Importing a self-signed leaf certificate from the console:

```
switch(config)# crypto pki certificate ss-leaf-cert
switch(config-cert-ss-leaf-cert)# import terminal self-signed
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-cert-import)# -----BEGIN CERTIFICATE-----
switch(config-cert-import)# MIID2TCCAsGgAwIBAgIJAKcrqokm6p9GMA0GCSqGSIb3DQEBCwUAM
switch(config-cert-import)# tDCCA5ygAwIBAgICEAEwDQYJKoZIhvcNAQELBQAwwYgxCzABAYTA1
switch(config-cert-import)# VQQGEWJVUzELMAkGA1UECAwCQ0ExDTALBgNVBACMBFJvc2UxDDAKB
...
switch(config-cert-import)# +fWQLxhp+jKJGZGOZz/FENt2uSfZHxlXiu8n3g+EgqExenY1pBRJr
switch(config-cert-import)# VuEEoNb/YfkPXHHva4Zfx223q+f694wlVsHkENSzqr2goHpa2fOzq
switch(config-cert-import)# alewwdmVqCES+x8bvhf3C/6IB6ePkEsnMlHNTeM=
switch(config-cert-import)# -----END CERTIFICATE-----
switch(config-cert-import)# -----BEGIN ENCRYPTED PRIVATE KEY-----
switch(config-cert-import)# MIIFDjBABgkqhkiG9w0BBQ0wMzAbBgkqhkiG9w0BBQwwDgQIT8Ni3
switch(config-cert-import)# MBQGCCqGSIb3DQMHBAiBHrejkdpcdASCBMjVxrrYYPnt3V1abr9k8
switch(config-cert-import)# 5GE0U99awh9ys4360WR95xOFGThvjkTyRWG511nGwVeLZs/7TPXWI
```

```
...
switch(config-cert-import)# hzc5ZT/w2F08icRI5mFbGoTAAw9IIWMOXGweaWQJDyKGrhg89GrnV
switch(config-cert-import)# M2UuP/tYuuO328QcenKZEJmZKCbX78oFRR+pgma4oeMaFTIyXE6Pr
switch(config-cert-import)# GAdCK8tkDiJ9DKbqdM5W0/nTJfqwUQlfl27dNrBAodsHdwr3UR99H
switch(config-cert-import)# SPo=
switch(config-cert-import)# -----END ENCRYPTED PRIVATE KEY-----
switch(config-cert-import)#
Enter import password: *****
Leaf certificate is validated as self-signed certificate and imported successfully.
switch(config-cert-ss-leaf-cert)#
```

Importing a leaf certificate from a remote file:

```
switch(config)# crypto pki certificate ss-leaf-cert2
switch(config-cert-ss-leaf-cert2)# import tftp://1.1.1.2/ss2.p12 self-signed
% Total      % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total     Spent    Left     Speed
100 3230 100 3230    0      0   875k      0 --:--:-- --:--:-- --:--:--   875k
100 3230 100 3230    0      0   831k      0 --:--:-- --:--:-- --:--:--   831k
Enter import password: *****
Leaf certificate is validated as self-signed certificate and imported successfully.
switch(config-cert-ss-leaf-cert2)#
```

## key-type

### Syntax

```
key-type {rsa [key-size <K-SIZE>] | ecdsa [curve-size <C-SIZE>]}
```

### Description

Sets the key type and key size for the current leaf certificate. The key type of the default certificate `local-cert` cannot be changed.

### Command context

```
config-cert-<CERT-NAME>
```

### Parameters

`rsa`

Specifies the key type as RSA.

`key-size <K-SIZE>`

Specifies the RSA key size in bits. Supported values: 2048, 3072, 4096. Default: 2048

`ecdsa`

Specifies the key type as ECDSA.

`curve-size <C-SIZE>`

Specifies the ECDSA elliptic curve size in bits. Supported values: 256, 348, 521. Default: 256

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Setting RSA encryption on the leaf certificate **leaf-cert**:

```
switch(config)# crypto pki certificate leaf-cert  
switch(config-cert-leaf-cert)# key-type rsa key-size 3072
```

Setting ECDSA encryption on the leaf certificate **leaf-cert**:

```
switch(config)# crypto pki certificate leaf-cert  
switch(config-cert-leaf-cert)# key-type ecdsa curve-size 521
```

## ocsp disable-nonce

### Syntax

```
ocsp disable-nonce  
no ocsp disable-nonce
```

### Description

Configures exclusion of the nonce from OCSF requests. A nonce is a unique identifier that an OCSF client inserts in an OCSF request and expects the OCSF responder to include it in the corresponding OCSF response. The nonce mechanism helps prevent replay attacks in which a malicious player attempts to masquerade as the OCSF responder. Although the nonce is included by default, it can be excluded. Some OCSF responders choose to not support the use of the nonce due to performance considerations.

The **no** form of this command re-enables nonce inclusion in OCSF requests.

### Command context

```
config-ta-<TA-NAME>
```

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Disable inclusion of the nonce in OCSF requests for TA profile **root-cert**:

```
switch(config)# crypto pki ta-profile root-cert  
switch(config-ta-root-cert)# ocsp disable-nonce
```

Enable inclusion of the nonce in OCSF requests for TA profile **root-cert**:

```
switch(config)# crypto pki ta-profile root-cert  
switch(config-ta-root-cert)# no ocsp disable-nonce
```

## ocsp enforcement-level

### Syntax

```
ocsp enforcement-level {strict | optional}  
no enforcement-level
```

### Description

Sets either strict or reduced enforcement of the OCSF check of certificates. Strict enforcement is enabled by default.

The `no` form of this command resets enforcement to its default of `strict`.

## Command context

`config-ta-<TA-NAME>`

## Parameters

`strict`

Sets strict OCSF checking of certificates. The certificate is accepted only if all possible checking (including validation failures, software system errors, configuration errors, transactional errors) is successful.

`optional`

Sets reduced OCSF checking of certificates. The certificate is accepted unless one or more of these validation errors occur:

- Response signature invalid.
- Nonce in response mismatch.
- Certificate revoked, but only when revocation checking is possible. If revocation check is not possible, the certificate is still accepted if there are no other validation errors.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Setting reduced OCSF checking of certificates:

```
switch(config)# crypto pki ta-profile root-cert
switch(config-ta-root-cert)# ocsf enforcement-level optional
```

Setting strict OCSF checking of certificates:

```
switch(config)# crypto pki ta-profile root-cert
switch(config-ta-root-cert)# ocsf enforcement-level strict
```

## ocsf url

### Syntax

```
ocsf url {primary | secondary} <URL>
no ocsf url {primary | secondary}
```

### Description

Configures the OCSF responder URLs that the current TA profile uses to verify the revocation status of an X.509 digital certificate. These URLs override the OCSF responder URL contained within the peer certificate being verified (as well as URLs defined in any intermediate CAs in the chain of trust).

If no OCSF responder URLs are defined for a TA profile (default setting), then the OCSF responder URL in the peer certificate is used for revocation status checking. (The OCSF responder URL is contained in a certificate's Authority Information Access field, which is an X.509 v3 certificate extension.)

The `no` form of this command deletes the specified OCSF responder URL (primary or secondary) from the current TA profile.

## Command context

config-ta-<TA-NAME>

## Parameters

{primary | secondary} <URL>

Specify the HTTP URL of the primary or secondary OCSF responder using either a fully qualified domain name or IPv4 address.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Defining the primary OCSF URL for the TA profile **root-cert**:

```
switch(config)# crypto pki ta-profile root-cert
switch(config-ta-root-cert)# revocation-check ocsf
switch(config-ta-root-cert)# ocsf url primary http://ocsf-server.site.com
```

Removing the primary OCSF URL from the TA profile **root-cert**:

```
switch(config)# crypto pki ta-profile oot-cert
switch(config-ta-root-cert)# revocation-check ocsf
switch(config-ta-root-cert)# no ocsf url primary
```

## ocsf vrf

### Syntax

```
ocsf vrf <VRF-NAME>
no ocsf vrf
```

### Description

Sets the VRF that the switch uses to communicate with OCSF responders for OCSF checking. VRF mgmt is used by default.

The **no** form of this command resets the VRF to its default **mgmt**.

### Command context

config-ta-<TA-NAME>

## Parameters

<VRF-NAME>

Specifies the name of the VRF the switch uses to communicate with OCSF responders. Default: **mgmt**.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Setting the OCSF responder VRF to **corp1**:

```
switch(config)# crypto pki ta-profile root-cert  
switch(config-ta-root-cert)# ocsp vrf corp1
```

Reverting the OCSF responder VRF to its default:

```
switch(config)# crypto pki ta-profile root-cert  
switch(config-ta-root-cert)# no ocsp vrf
```

## revocation-check ocsp

### Syntax

```
revocation-check ocsp  
no revocation-check
```

### Description

Enables certificate revocation checking for the current profile using the online certificate status protocol (OCSP).

The **no** form of this command disables certificate revocation checking for the current profile.

### Command context

config-ta-<TA-NAME>

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Enabling revocation checking for the TA profile **root-cert**:

```
switch(config)# crypto pki ta-profile root-cert  
switch(config-ta-root-cert)# revocation-check ocsp
```

Disabling revocation checking for the TA profile **root-cert**:

```
switch(config)# crypto pki ta-profile root-cert  
switch(config-ta-root-cert)# no revocation-check
```

## show crypto pki application

### Syntax

```
show crypto pki application
```

### Description

Shows certificate information for all features (applications) using leaf certificates that are managed by PKI.

### Command context

Manager (#)

### Authority

Administrators or local user group members with execution rights for this command.

## Examples

Showing certificate information for all features (applications) using leaf certificates:

```
switch# show crypto pki application1
```

Associated Applications	Certificate Name	Cert Status
https-server		not configured, using local-cert
syslog-client	local-cert	valid
hsc	xhscert	invalid, using local-cert
radsec-client	device-identity	valid

## show crypto pki certificate

### Syntax

```
show crypto pki certificate [<CERT-NAME> [plaintext | pem]]
```

### Description

Shows a list of all configured leaf certificates, or detailed information for a specific leaf certificate.

Possible values for Cert Status are: CSR pending, expired, expires soon, installed, malformed, not yet known.

Possible values for EST Status are: enroll failed, enroll pending, enroll retrying, enroll success, n/a (certificate is not EST-enrolled), reenroll failed, reenroll pending, reenroll retrying.

### Command context

Manager (#)

### Parameters

<CERT-NAME>

Specifies the leaf certificate name. Range: 1 to 32 alphanumeric characters excluding ".

plaintext

Shows certificate information in plain text.

pem

Shows certificate information in PEM format.

### Authority

Administrators or local user group members with execution rights for this command.

## Examples

Showing a list of all configured leaf certificates:

```
switch# show crypto pki certificate
```

Certificate Name	Cert Status	EST Status	Associated Applications
local-cert	installed	n/a	radsec-client, captive-portal
device-identity	installed	n/a	none
pod01-99-1	installed	n/a	https-server, est-client
syslog-1	CSR pending	enroll retrying	syslog-client

leaf-cert1	installed	enroll success	none
leaf-cert2	CSR pending	enroll failed	none

Showing detailed information (in plaintext format) for leaf certificate pod01-99-1:

```
switch# show crypto pki certificate pod01-99-1 plaintext

Certificate Name: pod01-99-1
Associated Applications:
  https-server, est-client
Certificate Status: installed
EST Status: n/a
Certificate Type: regular
Intermediates:
  Subject: C = US, ST = CA, O = Company, OU = Lab-IT, CN = DeviceCA
  Issuer: C = US, ST = CA, O = Company, OU = Lab-IT, CN = Lab-CA
  Serial Number: 0x02
  Subject: C = US, ST = CA, O = Company, OU = Lab-IT, CN = Lab-CA
  Issuer: C = US, ST = CA, O = Company, OU = Lab-IT, CN = Lab-Root
  Serial Number: 0x01
Certificate:
  Data:
    Version: 1 (0x0)
    Serial Number: 14529416756121781768 (0xc9a2db8f3e3f4608)
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: C=US, ST=CA, OU=Lab-IT, O=Company, CN=DeviceCA
    Validity
      Not Before: Jan 12 23:36:57 2018 GMT
      Not After : Nov 1 23:36:57 2020 GMT
    Subject: C=US, ST=CA, OU=Lab-IT, O=Company, CN=pod01-99-1
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      Public-Key: (2048 bit)
      Modulus:

          00:a0:cd:ef:1b:f9:b8:bd:39:fc:7a:0e:00:17:ff:
          2b:72:d8:4e:d4:df:49:36:ca:3a:f9:05:05:d7:e3:
          d1:97:29:71:e6:33:b8:bb:8e:f0:ee:a6:e4:4a:f8:
          ...
          fe:dd:d9:a0:af:59:47:25:b4:34:06:af:03:1d:33:
          30:c3:85:fe:5c:e7:19:7f:ff:3a:b2:21:b8:e8:ed:
          83:09
      Exponent: 65537 (0x10001)
    Signature Algorithm: sha256WithRSAEncryption
      39:f6:03:86:03:d9:05:61:39:25:5f:0d:75:cc:05:ae:04:7e:
      4c:a3:13:0b:f0:1e:af:68:0e:40:9f:ed:48:b6:5e:56:8c:53:
      46:5b:c9:a4:e0:b0:bc:31:4b:a7:5d:0a:ed:7c:9c:f6:bf:1e:
      ...
      39:f5:26:58:68:e2:13:ec:94:ac:60:8e:4b:b0:ba:45:cf:d6:
      6a:4b:9f:7d:ae:3f:e5:2e:81:fe:ac:b3:65:44:35:47:a5:2f:
      89:e7:58:a0
```

Showing detailed information (in PEM format) for leaf certificate leaf-cert1 with a status of CSR pending:

```
switch# show crypto pki certificate leaf-cert1 pem

Certificate Name: leaf-cert1

Associated Applications:
```



```

syslog-client
Certificate Status: CSR pending
EST Status: enroll retrying
Certificate Type: regular
-----BEGIN CERTIFICATE REQUEST-----

MIICtTCCAZ0CAQAwcDEWMBQGA1UEAxMNC3lzbG9nLTg0MBYGA1UECxMPQ
XJlYmEtUm9zZXZpbGx1MQ4wDAYDVQQKEyTESMBAGA1EBxMJUm9zZXZpbG
x1MQswCQYDVQQIEwJDQTELMAGA1UEBhMCVVMwggEiMSIb3DQEBAQUAA4I
...
cw2ytN6Idgh81k59x6DH7V/eORaKd5lq+o07nkr6+QBf5L3f5Kb+TOFio
lei+EdCHMxxc07MK0n3dkziSW25HFUGsyEXVMK+BiD3zbKDoUe6XVhvqI
mamXyghigLYDcbsn6WVw==
-----END CERTIFICATE REQUEST-----

```

## show crypto pki ta-profile

### Syntax

```
show crypto pki ta-profile [<TA-NAME>]
```

### Description

Shows a list of all configured TA profiles, or detailed information for a specific profile.



This command shows information for both directly-configured TA profiles and TA profiles that were dynamically downloaded from EST servers.

### Command context

Manager (#)

### Parameters

<TA-NAME>

Specifies the TA profile name. Range: 1 to 48 alphanumeric characters excluding ".".

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Showing a list of all configured TA profiles:

```

switch# show crypto pki ta-profile

Profile Name          TA Certificate          Revocation Check
-----
BASE_CA               Installed,valid         disabled
BASE02_CA            Installed,expired       disabled
root-cert             Installed,valid         OCSP
ROOT-A_CA            Not Installed           OCSP
EST-Service1          Installed,valid         None
EST-Service2          Installed,valid         None

```

Showing detailed information for TA profile **root-cert**:

```

switch# show crypto pki ta-profile root-cert

TA Profile Name           : root-cert
Revocation Check          : OCSF
  OSCP Primary URL        : http://ocsp1.domain.com
  OSCP Secondary URL      : Not Configured
  OSCP Disable-nonce      : false
  OSCP Enforcement Level  : strict
  OSCP VRF                : mgmt
TA Certificate: Installed and valid
Version: 3 (0x2)
Serial Number:
  74:e6:6d:22:3f:52:cc:94:43:41:ab:66:a8:8d:47:b1
Signature Algorithm: sha1withRSAEncryption
Issuer: OU=DeviceTrust, OU=Operations, O=Site, C=US,
       CN=Site Trusted Computing Root CA 1.0
Validity
  Not Before: Sep 14 03:12:06 2007 GMT
  Not After : Sep 14 03:21:14 2032 GMT
Subject: OU=DeviceTrust, OU=Operations, O=Site, C=US,
       CN=Site Trusted Computing Root CA 1.0
Subject Public Key Info:
  Public Key Algorithm: rsaEncryption
  RSA Public Key: (2048 bit)
  Modulus (2048 bit):
    30:0d:06:09:2a:86:48:86:f7:0d:01:01:01:05:33:
    03:82:01:0f:00:30:82:01:3a:02:82:01:01:00:ac:
    3d:60:3a:2e:ca:a4:34:db:5c:3b:6b:07:df:73:62:
    ...
    20:c8:df:63:14:5a:e8:d3:ea:83:d8:47:a3:b5:2e:
    bb:64:51:f0:be:13:b6:91:e4:32:45:58:5e:1f:0d:
    02:03:01:00:01
  Exponent: 65537 (0x10001)
X509v3 extensions:
  X509v3 Key Usage:
    Digital Signature, Certificate Signing, CRL Signing
  X509v3 Basic Constraints:
    CA:TRUE, pathlen:4
  X509v3 Subject Key Identifier:
    eb:d7:ec:db:8a:cb:f2:51:d5:06:e1:42:7b:39:a7:d0:1e:31:6e:bf

Signature Algorithm: sha1withRSAEncryption
  1c:90:f3:a4:f0:0d:e2:e3:e9:ae:01:e1:7d:a7:13:e2:cc:0b:
  17:31:26:92:a2:5d:1d:19:60:54:03:13:9b:e1:73:6c:e4:b3:
  01:4f:4e:ae:61:bd:ae:b6:12:d3:ab:08:ae:8c:47:92:d7:0d:
  ...
  ca:cf:11:78:55:6d:06:49:fa:d4:8d:f3:ef:7f:79:38:35:5d:
  16:5a:57:7f:a8:dc:b0:f8:a2:04:0d:17:0b:bb:58:32:30:e0:
  2d:a8:37:a2

```

## ta-certificate

### Syntax

```
ta-certificate { [import [terminal]] | import {<REMOTE-URL> | <STORAGE-URL>} }
```

### Description

Imports a CA certificate for use in the current TA profile. The certificate must be in PEM format. The PEM data must be delimited with these lines:

```

-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----

```




---

Only the first certificate in the PEM data is imported. Any additional certificates are ignored.

---

## Command context

config-ta-<TA-NAME>

## Parameters

[import [terminal]]

Import the certificate by pasting at the console (the default). This form of importing is selected whether `ta-certificate` is entered without parameters or if only `import` is entered or if `import terminal` is entered. Upon execution, the `config-ta-cert` context is entered for certificate pasting. To complete certificate data entry press Control-D in your terminal program. Alternatively, the pasted certificate data can include at its end the delimiter `END_OF_CERTIFICATE` (after the `-----END CERTIFICATE-----` line), making entry of Control-D unnecessary..

import <REMOTE-URL>

Import the certificate from a file on a remote TFTP or SFTP server. The URL syntax is:

{tftp:// | sftp://<USER>@} {<IP>|<HOST>} [:<PORT>] [:blocksize=<SIZE>]/<FILE>

import <STORAGE-URL>

Available on switch families that provide USB device file import capability, import the certificate from a file on a USB storage device inserted in the switch USB port. The URL syntax is:

usb:/<FILE>

## Authority

Administrators or local user group members with execution rights for this command.

## Example

Importing a certificate into the TA profile **root-cert** by pasting PEM-format certificate data at the console:

```
switch(config)# crypto pki ta-profile root-cert
switch(config-ta-root-cert)# ta-certificate import terminal
Paste the certificate in PEM format below, then hit enter and ctrl-D:
switch(config-ta-cert)# -----BEGIN CERTIFICATE-----
switch(config-ta-cert)# MIIDuTCCAqECCQCuoxeJ2ZNYcjANBgkqhkiG9w0BAQsFADCBQzELMAEBh
switch(config-ta-cert)# VVMxEzARBgNVBAgMCKNhGlmb3JuaWExEDAOBgNVBACMB1JvY2tsDAKBg
switch(config-ta-cert)# BAoMA0hQTjEVMBMGA1UECwwMSFB0Um9zZXZpbGx1MSowKAYDVQOCG5zd
...
switch(config-ta-cert)# x3WFf3dFZ8o9sd5LVAHneH/ztb9MP34z+le1V346r12L2kpxmTOVJVyTO
switch(config-ta-cert)# BIzD/ST/HaWI+0S+S80rm93PSscEbb9GWk7vshh5EnW/moehBKcE40lzy
switch(config-ta-cert)# 3LvMLZcssSe5J2Ca2XIhfDme8UaNZ7syGYMsAW0nG7yYHWkEOQu9s
switch(config-ta-cert)# -----END CERTIFICATE-----
switch(config-ta-cert)#
The certificate you are importing has the following attributes:
Issuer: C=US, ST=CA, L=Rocklin, O=Company, OU=Site,
       CN=site.com/emailAddress=test.ca@site.com
Subject: C=US, ST=CA, L=Rocklin, O=Company, OU=Site,
       CN=9000/emailAddress=test.ca@site.com
Serial Number: 12121221634631568498 (0xae51217d5945772)

TA certificate import is allowed only once for a TA profile
Do you want to accept this certificate (y/n)? y
TA certificate accepted.
switch(config-ta-root-cert)#
```

Importing a certificate into the TA profile **root-cert2** from file `rcert2-data` on the USB device:

```

switch(config)# crypto pki ta-profile root-cert2
switch(config-ta-root-cert2)# ta-certificate import usb:/rcert2-data
The certificate you are importing has the following attributes:
Issuer: C=US, ST=California, L=Rocklin, O=Company, OU=Site,
CN=site.com/emailAddress=test.ca@site.com
Subject: C=US, ST=California, L=Rocklin, O=Company, OU=Site,
CN=9000/emailAddress=test.ca@site.com
Serial Number: 12121221634631568498 (0xae51217d5945772)

TA certificate import is allowed only once for a TA profile
Do you want to accept this certificate (y/n)? y
TA certificate accepted.
switch(config-ta-root-cert2)#

```

## subject

### Syntax

```

subject [common-name <COMMON-NAME>] [country <COUNTRY>] [locality <LOCALITY>]
[org <ORG-NAME>] [org-unit <ORG-UNIT>] [state <STATE>]

```

### Description

Sets the subject fields for the current leaf certificate. If the `common-name` parameter is not specified, then you are prompted to define a value for each field. If a configured value exists for any field, it is presented as the default.

The subject fields of the default certificate `local-cert` cannot be changed.

### Command context

```
config-cert-<CERT-NAME>
```

### Parameters

```
common-name <COMMON-NAME>
```

Specifies the common name.

```
country <COUNTRY>
```

Specifies the region.

```
locality <LOCALITY>
```

Specifies the locality.

```
org <ORG-NAME>
```

Specifies the organization.

```
org-unit <ORG-UNIT>
```

Specifies the organizational unit.

```
state <STATE>
```

Specifies the state.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Setting subject fields for the leaf certificate **leaf-cert**:

```
switch(config-cert-leaf-cert)# subject common-name Leaf01 country US  
locality CA org Company org-unit Site state CA
```

Setting subject fields for the leaf certificate **leaf-cert** interactively:

```
switch(config-cert-leaf-cert)# subject  
Do you want to use the switch serial number as the common name (y/n)? n  
Enter Common Name : Leaf01  
Enter Org Unit : Site  
Enter Org Name : Company  
Enter Locality : Rocklin  
Enter State : CA  
Enter Country : US  
switch(config-cert-leaf-cert)#
```

## PKI EST commands

### arbitrary-label

#### Syntax

```
arbitrary-label <LABEL>  
no arbitrary-label
```

#### Description

Within the EST profile context, configures the generic optional label (also known as arbitrary label) to be concatenated to the EST server URL that is configured with the `url` command. There is no arbitrary label configured by default. Any existing arbitrary label is replaced by this command. The use of arbitrary labels is optional.

RFC 7030 allows the use of arbitrary labels so that one EST server may serve multiple CAs with the same server URL that gets concatenated with different arbitrary labels. The same label is used for every request made under a particular EST profile.

Some EST schemes use arbitrary labels in a more sophisticated way, defining different labels for different types of requests under the same EST profile. For example, the CA certificate request could use the generic label (configured with this `arbitrary-label` command), the certificate enrollment request could use the enrollment label (configured with the `arbitrary-label-enrollment` command), and the re-enrollment request could use the re-enrollment label (configured with the `arbitrary-label-reenrollment` command). Note that only one label of each of the three available types can be configured in any EST profile.

The no form of this command removes the generic arbitrary label.

#### Command context

```
config-est-<EST-NAME>
```

#### Parameters

<LABEL>

Specifies the generic arbitrary label. Range: Up to 64 characters.

#### Authority

Administrators or local user group members with execution rights for this command.

#### Examples

Configuring the URL and generic arbitrary label. Note that with the URL and arbitrary label configured in this example, the final URL the switch uses to request CA certificates from the EST server is `https://est-service999.com/.well-known/est/rsa2048/cacerts`.

```
switch(config)# crypto pki est-profile EST-service1
switch(config)# url https://est-service999.com/.well-known/est
switch(config-est-EST-service1)# arbitrary-label rsa2048
```

Removing the generic arbitrary label:

```
switch(config)# crypto pki est-profile EST-service1
switch(config-est-EST-service1)# no arbitrary-label
```

## arbitrary-label-enrollment

### Syntax

```
arbitrary-label-enrollment <LABEL>
no arbitrary-label-enrollment
```

### Description

Within the EST profile context, configures the arbitrary enrollment label to be concatenated to the EST server URL that is configured with the `url` command. This label is specific to the enrollment operation. There is no arbitrary enrollment label configured by default. Any existing arbitrary enrollment label is replaced by this command. The use of arbitrary enrollment labels is optional.

When the enrollment label is not configured, the generic arbitrary label (created with the `arbitrary-label` command) is used (if configured) for enrollment.

RFC 7030 allows the use of arbitrary labels so that one EST server may serve multiple CAs with the same server URL that gets concatenated with different arbitrary labels. The same label is used for every request made under a particular EST profile.

Some EST schemes use arbitrary labels in a more sophisticated way, defining different labels for different types of requests under the same EST profile. For example, the CA certificate request could use the generic label (configured with the `arbitrary-label` command), the certificate enrollment request could use the enrollment label (configured with this `arbitrary-label-enrollment` command), and the re-enrollment request could use the re-enrollment label (configured with the `arbitrary-label-reenrollment` command). Note that only one label of each of the three available types can be configured in any EST profile.

The no form of this command removes the arbitrary enrollment label.

### Command context

```
config-est-<EST-NAME>
```

### Parameters

<LABEL>

Specifies the arbitrary enrollment label. Range: Up to 64 characters.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Configuring the arbitrary enrollment label:

```
switch(config)# crypto pki est-profile EST-service1  
switch(config-est-EST-service1)# arbitrary-label-enrollment ipsec-v7
```

Removing the arbitrary enrollment label :

```
switch(config)# crypto pki est-profile EST-service1  
switch(config-est-EST-service1)# no arbitrary-label-enrollment
```

## arbitrary-label-reenrollment

### Syntax

```
arbitrary-label-reenrollment <LABEL>  
no arbitrary-label-reenrollment
```

### Description

Within the EST profile context, configures the arbitrary re-enrollment label to be concatenated to the EST server URL that is configured with the `url` command. This label is specific to the re-enrollment operation. There is no arbitrary re-enrollment label configured by default. Any existing arbitrary re-enrollment label is replaced by this command. The use of arbitrary re-enrollment labels is optional.

When the re-enrollment label is not configured, the generic arbitrary label (created with the `arbitrary-label` command) is used (if configured) for re-enrollment.

RFC 7030 allows the use of arbitrary labels so that one EST server may serve multiple CAs with the same server URL that gets concatenated with different arbitrary labels. The same label is used for every request made under a particular EST profile.

Some EST schemes use arbitrary labels in a more sophisticated way, defining different labels for different types of requests under the same EST profile. For example, the CA certificate request could use the generic label (configured with the `arbitrary-label` command), the certificate enrollment request could use the enrollment label (configured with the `arbitrary-label-enrollment` command), and the re-enrollment request could use the re-enrollment label (configured with this `arbitrary-label-reenrollment` command). Note that only one label of each of the three available types can be configured in any EST profile.

The no form of this command removes the arbitrary re-enrollment label.

### Command context

```
config-est-<EST-NAME>
```

### Parameters

<LABEL>

Specifies the arbitrary re-enrollment label. Range: Up to 64 characters.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Configuring the arbitrary re-enrollment label:

```
switch(config)# crypto pki est-profile EST-service1  
switch(config-est-EST-service1)# arbitrary-label-reenrollment ipsec-v7
```

Removing the arbitrary re-enrollment label :

```
switch(config)# crypto pki est-profile EST-service1  
switch(config-est-EST-service1)# no arbitrary-label-reenrollment
```

## crypto pki est-profile

### Syntax

```
crypto pki est-profile <EST-NAME>  
no crypto pki est-profile <EST-NAME>
```

### Description

Creates a certificate Enrollment over Secure Transport (EST) profile and changes to the `config-est-<EST-NAME>` context for the profile. Each EST profile stores information about the EST service, including EST server URL. Up to 16 profiles can be created.

If the specified EST profile exists, this command changes to the `config-est-<EST-NAME>` context for the profile.

The `no` form of this command deletes the specified EST profile. It also deletes the TA profiles whose CA certificates were downloaded from the corresponding EST server, and the leaf certificates that were enrolled using this EST profile.



---

The deletion of the related TA profiles and enrolled certificates is permanent. If the EST profile is in the startup configuration and the EST profile is deleted but this deletion is not updated in the startup configuration before a switch reboot, the EST profile will still exist after the reboot but the related TA profiles and enrolled certificates will not exist.

---

### Command context

config

### Parameters

<EST-NAME>

Specifies the EST profile name. Range: Up to 32 alphanumeric characters (excluding ").

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Creating EST profile **EST-Service1**:

```
switch(config)# crypto pki est-profile EST-Service1  
switch(config-est-service1)#
```

Removing EST profile **service1**:

```
switch(config)# no crypto pki est-profile EST-Service1
```

## enroll est-profile



## Syntax

```
enroll est-profile <EST-NAME>
```

## Description

Enrolls a leaf certificate through a remote EST (Enrollment over Secure Transport) server.

Per RFC 7030, EST enables clients to request certificate signing services over secure TLS connections. The switch generates a key pair and the corresponding CSR. The CSR is sent to the EST server to request signing, and the signed certificate is returned to the switch where it is validated. If the whole process succeeds, the certificate can be used as a leaf certificate on the switch. When the leaf certificate approaches its expiry date, it will be renewed automatically through the same EST server.

Each enrollment or re-enrollment attempt starts with a `/cacerts` request sent to the EST server to get the latest chain of CA certificates. After the enrollment or re-enrollment succeeds, this chain of CA certificates will be compared with those downloaded previously from the same EST server. Updates will be made as appropriate.

The subject fields of the current leaf certificate must be defined before running this command. If the common name subject field is not configured, this command is rejected.

This command cannot be used to enroll or renew the default certificate "local-cert."

## Command context

```
config-cert-<CERT-NAME>
```

## Parameters

<EST-NAME>

Specifies an existing EST profile name. Range: Up to 32 alphanumeric characters (excluding ").

## Authority

Administrators or local user group members with execution rights for this command.

## Example

Enrolling leaf certificate **leaf-cert1** through the EST server identified in EST profile `EST-service1`:

```
switch(config-cert-leaf-cert1)# enroll est-profile EST-service1
You are enrolling a certificate with the following attributes:
  Subject: C=US, ST=CA, L=Roseville, OU=Aruba-Roseville, O=Aruba,
          CN=leaf-cert1
  Key Type: RSA (2048 bits)

Continue (y/n)? y
Certificate enrollment via EST-service1 has been initiated.
Please use `show crypto pki certificate leaf-cert1` to check its status.

switch(config-cert-leaf-cert1)#
```

## reenrollment-lead-time

### Syntax

```
reenrollment-lead-time <LEAD-TIME>
no reenrollment-lead-time
```

### Description

Within the EST profile context, sets the certificate re-enrollment lead time which is the number of days before certificate expiry date that certificate re-enrollment will be initiated.

The no form of this command resets the EST server re-enrollment lead time to its default of 2 days.

## Command context

config-est-*<EST-NAME>*

## Parameters

*<LEAD-TIME>*

Specifies the certificate re-enrollment lead time in days. Range: 0 to 30 days. Default: 2 days.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Setting the certificate re-enrollment lead time to 15 days:

```
switch(config)# crypto pki est-profile EST-service1
switch(config-est-EST-service1)# reenrollment-lead-time 15
```

Resetting the certificate re-enrollment lead time to its default of 2 days :

```
switch(config)# crypto pki est-profile EST-service1
switch(config-est-EST-service1)# no reenrollment-lead-time
```

## retry-count

### Syntax

```
retry-count <RETRIES>
no retry-count
```

### Description

Within the EST profile context, sets the maximum number of retries to be attempted after the initial certificate enrollment request fails.

The no form of this command resets the maximum number of certificate enrollment request retries to its default of 3.

## Command context

config-est-*<EST-NAME>*

## Parameters

*<RETRIES>*

Specifies the maximum number of certificate enrollment request retries. Range: 0 to 32 retries. Default: 3 retries.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Setting the retry count to 5 retries:

```
switch(config)# crypto pki est-profile EST-service1  
switch(config-est-EST-service1)# retry-count 5
```

Resetting the retry count to its default of 3 retries:

```
switch(config)# crypto pki est-profile EST-service1  
switch(config-est-EST-service1)# no retry-count
```

## retry-interval

### Syntax

```
retry-interval <INTERVAL>  
no retry-interval
```

### Description

Within the EST profile context, sets the interval at which a failed certificate enrollment request is retried. The no form of this command resets the enrollment request retry interval to its default of 30 seconds.

### Command context

config-est-<EST-NAME>

### Parameters

<INTERVAL>

Specifies the enrollment request retry interval in seconds. Range: 30 to 600 seconds. Default: 30 seconds.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Setting the certificate enrollment request retry interval to 45 seconds:

```
switch(config)# crypto pki est-profile EST-service1  
switch(config-est-EST-service1)# retry-interval 45
```

Resetting the retry interval to its default of 30 seconds:

```
switch(config)# crypto pki est-profile EST-service1  
switch(config-est-EST-service1)# no retry-interval
```

## show crypto pki est-profile

### Syntax

```
show crypto pki est-profile [<EST-NAME>]
```

### Description

Shows a list of all configured EST profiles, or detailed information for a specific profile.

## Command context

Manager (#)

## Parameters

<EST-NAME>

Specifies the EST profile name. Range: Up to 32 alphanumeric characters excluding ".".

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Showing a list of all configured EST profiles:

```
switch# show crypto pki est-profile
```

Profile Name	Downloaded TA Profiles	Enrolled Certificates
EST-service1	2	3
EST-service2	1	2
EST-service3	2	0

Showing detailed information for EST profile **EST-service1**:

```
switch# show crypto pki est-profile EST-service1
Profile Name           : EST-service1
Service VRF            : mgmt
Service URL            : https://est-service999.com
Arbitrary Label        : not configured
Arbitrary Label Enrollment : /ipsec-VP7
Arbitrary Label Reenrollment : not configured
Authentication Username : est1
Authentication Password :
  AQBapREALpWYm2z7L1LanOtR3vGkqhBN1hBUU2CuvQXUF/ggYgAAAnAnGTnKq49P4c
  dNQ6UqPbjHL4XzCO0T04djkhsUXPKGfnsWuFEONveh+JbEobqKImfwJjc3eWHiaUb
  eNpPx2zN2Q1DdyxAAQi4rmKr8LITMTTmd7qr
Retry Interval         : 45 seconds
Retry Count            : 5 times
Reenrollment Lead Time : 2 days
Downloaded TA Profiles : 2
Enrolled Certificates  :
  leaf-cert1
  leaf-cert2
  leaf-cert3
```

## url

### Syntax

```
url <URL>
no url
```

### Description

Within the EST profile context, configures the URL of the certificate enrollment EST server. This is not configured by default. Any existing URL is replaced by this command.

The no form of this command removes the EST server URL within the selected EST profile. The removal of the URL does not affect the TA profiles and enrolled certificates from the EST server.

## Command context

config-est-<EST-NAME>

## Parameters

<URL>

Specifies the EST server URL. Range: Up to 192 characters.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

- The configuration and update of the EST profile URL triggers the sending of a `/cacerts` request to the EST server. A successful request will result in a chain of trusted CA certificates being downloaded from the EST server. Each CA certificate, either root CA certificates or intermediate CA certificates, will be saved as a TA profile, with TA profile name `<est-name>-est-taNN` with `NN` representing two numerical digits. This TA profile naming scheme with the `-est-taNN` suffix is reserved for TA profiles downloaded from EST servers.
- Upon connection with an EST server, the switch authenticates the server by validating the server certificate. For this validation to succeed, a TA profile needs to pre-exist in the switch with a CA certificate from the issuer chain of the server certificate. Once the server is authenticated, all CA certificates in its `/cacerts` response will be trusted, with no further validation occurring for them.
- The TA profiles with CA certificates downloaded from an EST server will have their revocation check set to OCSP, enforcement set to optional, and the OCSP VRF set to the same as that of the EST profile.

## Examples

Configuring the EST server URL :

```
switch(config)# crypto pki est-profile EST-service1
switch(config-est-EST-service1)# url https://est-service999.com/.well-known/est
```

Removing the EST server URL :

```
switch(config)# crypto pki est-profile EST-service1
switch(config-est-EST-service1)# no url
```

## username

### Syntax

username <USERNAME> password [ciphertext <CIPHERTEXT-PASSWORD> | plaintext <PLAINTEXT-PASSWORD>]

no username

### Description

Within the EST profile context, configures the user account information for the EST server that is used to authenticate the switch before accepting requests from the switch. This is not configured by default. Any existing username and password is replaced by this command.

When entered without either optional `ciphertext` or `plaintext` parameters, the plaintext password is prompted for twice, with the characters entered masked with "\*" symbols.

The no form of this command removes the user account information within the selected EST profile.

There are two ways the EST client on a CX switch can prove itself to an EST server: a certificate, and/or username and password. At least one of the two must be configured for the EST request to succeed. If both are configured, certificate authentication will be used. If a certificate is not configured or certificate authentication fails, and username and password is configured, the username and password will be sent to the EST server for authentication.

## Command context

`config-est-<EST-NAME>`

## Parameters

`<USERNAME>`

Specifies the EST server account user name. The exact user name requirements are set by the chosen EST service. Range: Up to 32 alphanumeric characters.

`ciphertext <CIPHERTEXT-PASSWORD>`

Specifies the EST server account password as Base64 ciphertext. No password prompts are provided and the ciphertext password is validated before the configuration is applied for the user.



---

The ciphertext password must be gotten from the EST service.

---

`plaintext <PLAINTEXT-PASSWORD>`

Specifies the password without prompting. The password is visible as cleartext when entered but is encrypted thereafter. The exact password requirements are set by the chosen EST service. Range: Up to 64 alphanumeric characters.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring an EST user with prompted cleartext password entry :

```
switch(config)# crypto pki est-profile EST-service1
switch(config-est-EST-service1)# username est1 password
Enter password: *****
Confirm password: *****
switch(config-est-EST-service1)#
```

Configuring an EST user with direct cleartext password entry:

```
switch(config)# crypto pki est-profile EST-service2
switch(config-est-EST-service2)# username est1 password plaintext concept_leap739
```

Configuring an EST user with ciphertext password entry :

```
switch(config)# crypto pki est-profile EST-service3
switch(config-est-EST-service3)# username est1 password ciphertext
AQBpRALpWYm2z7L1LanOtR3vGkqhN1hBU2CuvQXUF/ggYgAAAHWaPqxU6nAnGTnKq49P4cdNQ6U
qPbjHL4XzO0T04djkUPKGfnsWuFEONveh+JbEobq63+1k80qBKImfwJjc3eWHiaUbeNpPx2zN2Q
1DdyxAAQi4rmKr8LITMTMd7qr
```

Removing the EST user account information for EST profile EST-service2:

```
switch(config)# crypto pki est-profile EST-service2
switch(config-est-EST-service2)# no username
```

## vrf

### Syntax

```
vrf <VRF-NAME>
no vrf
```

### Description

Within the EST profile context, selects the VRF through which the EST server can be reached. Any existing VRF selection is replaced by this command. When this command is not used, VRF `mgmt` is used by default on switch families supporting the `mgmt` VRF, otherwise the default VRF named `default` is used.

The no form of this command selects the default VRF either `mgmt` or `default`.

### Command context

```
config-est-<EST-NAME>
```

### Parameters

<VRF-NAME>

Specifies the name of the VRF to use for EST server communication.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Selecting VRF `it-services` for EST server communications:

```
switch(config)# crypto pki est-profile EST-service1
switch(config-est-EST-service1)# vrf it-services
```

Resetting the VRF to its default of `mgmt` for EST server communications:

```
switch(config)# crypto pki est-profile EST-service1
switch(config-est-EST-service1)# no vrf
```



---

MACsec is exclusive to the 8360 Switch Series, model JL717A.

---

Media Access Control security (MACsec) :

- Provides Layer 2 security protecting network communications against a range of attacks including: denial of service, intrusion, man-in-the-middle, and eavesdropping. These attacks exploit Layer 2 vulnerabilities and often cannot be detected.
- Provides Layer 2 hop-by-hop encryption on point-to-point Ethernet links.
- Enables a bi-directional secure link after an exchange and verification of security keys between two connected devices.
- Secures switch-to-switch infrastructure using the MKA (MACsec Key Agreement) protocol and Static CAK (Connectivity Association Key).
- Is intended for wired LANs.

More specifically, MACsec provides:

- **Connectionless data integrity:** Unauthorized changes to data cannot be made without being detected. Each MAC frame carries a separate integrity verification code.
- **Data origin authenticity:** A received MAC frame is guaranteed to have been sent by the authenticated device.
- **Confidentiality:** The data payload of each MAC frame is encrypted to prevent it from being eavesdropped by unauthorized parties.
- **Replay protection:** MAC frames copied from the network by an attacker cannot be resent into the network without being detected.
- **Bounded receive delay:** MAC frames cannot be intercepted by a man-in-the-middle attack and delayed by more than a few seconds without being detected.

## MACsec configuration basics

A simple configuration example is provided here to illustrate MACsec configuration on the 8360 Switch Series:

- Creating and configuring a MACsec policy:

```
switch(config)# macsec policy MS_Policy1
switch(config-macsec-policy)# cipher-suite gcm-aes-256 gcm-aes-xpn-256
switch(config-macsec-policy)# replay-protection window-size 100
switch(config-macsec-policy)# exit
switch(config)#
```

- Creating and configuring an MKA policy:



```
switch(config)# mka policy MKA_Policy1
switch(config-mka-policy)# pre-shared-key ckn abcdef12 cak plaintext 123abcdef
switch(config-mka-policy)# key-server-priority 5
switch(config-mka-policy)# exit
switch(config)#
```

- Applying the MACsec and MKA policy to a port range:

```
switch(config)# interface 1/1/1-1/1/4
switch(config-if-<1/1/1-1/1/4>)# apply macsec policy MS_Policy1
switch(config-if-<1/1/1-1/1/4>)# apply mka policy MKA_Policy1
switch(config-if-<1/1/1-1/1/4>)# exit
switch(config)#
```

- Show commands are provided for showing policy information and monitoring MACsec and MKA status and statistics:

```
switch(config)# show macsec policy MS_Policy1
...
switch# show macsec status
...
switch# show macsec statistics
...
switch# show mka policy MKA_Policy1
...
switch# show mka status
...
switch# show mka statistics
...
```

## MACsec commands




---

MACsec is exclusive to the 8360 Switch Series, model JL717A.

---

### apply macsec policy

#### Syntax

```
apply macsec policy <MACSEC-POLICY-NAME>
no apply macsec policy
```

#### Description

Within the selected interface context, applies the specified MACsec policy to the selected port. When a MACsec policy is applied to a port, MACsec is enabled on the port and all data traffic is blocked on the port until a secure channel is successfully established.




---

A MACsec policy can be applied to a physical interface port that is not part of any LAG ports or to a lag port. It can also be applied to an interface that is configured as an MCLAG, VSX keep-alive, or VSX inter-switch-link.

---

If a MACsec policy is already applied to the selected port, this command replaces the existing policy application.



---

For MACsec to work, an MKA policy must also be configured and applied to the same ports.

---

The `no` form of this command dissociates the specified policy from the port.

## Command context

`config-if`

## Parameters

`<MACSEC-POLICY-NAME>`

Specifies the MACsec policy name. Range: 1 to 32 alphanumeric characters including only the three special characters "." (period), "-" (hyphen), and "\_" (underscore).

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

- When any MACsec or MKA policy parameter is updated, any active MACsec session on all interfaces running the MACsec or MKA policy is terminated and restarted. This is indicated with the following prompt that provides an opportunity to not execute the `apply` command.

```
This policy is currently in use by one or more interfaces.  
Updating the policy will cause existing MACsec sessions using  
the policy to restart.  
Continue (y/n)?
```

- For non-LAG ports, a range of ports can be specified in the `interface` command used to enter the interface context. For example, entering the interface context for ports 1/1/1 through 1/1/2:

```
switch(config)# interface 1/1/1-1/1/2  
switch(config-if-<1/1/1-1/1/2>)# apply macsec policy MS_Policy1
```

- Not all interfaces on a switch may support the MACsec capability. An error will be generated when a policy is applied to a physical interface that is not capable of MACsec. For LAG ports, any non-MACsec capable interfaces that are part of the LAG will be blocked.
- The 32-port 8360 Switch Series (model JL717A) does not support both MACsec and priority-based flow-control (PFC) on same interface. Applying a MACsec policy to an interface associated with an existing PFC configuration will disable the interface. PFC must be unconfigured on the interface before MACsec can be used.

## Examples

Applying a MACsec policy to a range of two ports:

```
switch(config)# interface 1/1/1-1/1/2  
switch(config-if-<1/1/1-1/1/2>)# apply macsec policy MS_Policy1
```

Attempting to apply a MACsec policy to a port that already has PFC enabled:

```
switch(config)# interface 1/1/3
switch(config-if)# apply macsec policy MS_Policy1

MACsec and priority-based flow control (PFC) cannot be configured at the same time
on
this interface. Applying a MACsec policy will disable the interface until PFC is
removed.
Continue (y/n)?
```

Attempting to apply a MACsec policy to a port that is not MACsec capable:

```
switch(config)# interface 1/1/5
switch(config-if)# apply macsec policy MS_Policy1

MACsec is not supported on the interface.
switch(config-if)#
```

Removing MACsec policy association from a port:

```
switch(config)# interface 1/1/1
switch(config-if)# no apply macsec policy
```

Applying a MACsec policy to a LAG port:

```
switch(config)# interface lag 1
switch(config-if)# apply macsec policy MS_Policy1
```

## cipher-suite

### Syntax

```
cipher-suite {<CIPHER-SUITE>} [<CIPHER-SUITE>] ... [<CIPHER-SUITE>]
no cipher-suite [<CIPHER-SUITE>] ... [<CIPHER-SUITE>]
```

### Description

Within the MACsec policy context, configures one or more cipher suites to be used to generate the SAK (Secure Authentication Key) for when the switch is the key server. When multiple cipher suites are configured, the most secure cipher suite is considered first during negotiation.

The no form of this command (without the <CIPHER-SUITE> parameter) resets to the default of considering (during negotiation) all supported cipher suites while giving priority to the most secure suite gcm-aes-xpn-256. Include the <CIPHER-SUITE> parameter to disable a particular cipher suite.

### Command context

config-macsec-policy

### Parameters

<CIPHER-SUITE>

Selects the cipher suite. Available cipher suites are:

- gcm-aes-128: AES-128 encryption with Galois/Counter mode.
- gcm-aes-256: AES-256 encryption with Galois/Counter mode.

- gcm-aes-xpn-128: AES-128 encryption with Galois/Counter mode and extended packet numbering.
- gcm-aes-xpn-256: AES-128 encryption with Galois/Counter mode and extended packet numbering. (The default and the most secure.)

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Enabling a single cipher suite:

```
switch(config-macsec-policy) # cipher-suite gcm-aes-128
```

Enabling two cipher suites:

```
switch(config-macsec-policy) # cipher-suite gcm-aes-256 gcm-aes-xpn-256
```

Disabling a particular cipher suite:

```
switch(config-macsec-policy) # no cipher suite gcm-aes-128
```

Resetting to the default of considering all available cipher suites while giving priority to gcm-aes-xpn-256:

```
switch(config-macsec-policy) # no cipher-suite
```

## clear macsec statistics

### Syntax

```
clear macsec statistics [interface <IF-RANGE>]
```

### Description

Clears MACsec statistics on all MACsec-enabled interfaces or on a specific interface or interface range. MACsec statistics are cleared for the entire switch rather than just in the current user session.

### Command context

Operator (>) or Manager (#)

### Parameters

```
interface <IF-RANGE>
```

Specifies one or more interfaces for which MACsec statistics information is to be cleared.

## Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

## Examples

Clearing MACsec statistics on an interface range:

```
switch# clear macsec statistics 1/1/1-1/1/4
```

Clearing MACsec statistics on all MACsec-enabled interfaces:

```
switch# clear macsec statistics
```

## confidentiality

### Syntax

```
confidentiality [offset {0|30|50}]  
no confidentiality
```

### Description

Within the MACsec policy context, enables Ethernet packet encryption after the MACsec header, optionally including a start-of-encryption offset. Confidentiality is enabled by default with an offset of 0 bytes after the MACsec header.

An offset of 0 causes the entire packet (after the MACsec header) to be encrypted. It is sometimes desirable to offset the start of the encryption deeper into the packet to allow for fields such as MPLS labels and 802.1Q tags to remain unencrypted.

Omitting the `offset` parameter enables confidentiality with whatever offset was configured previously.

The `no` form of this command disables confidentiality.

### Command context

```
config-macsec-policy
```

### Parameters

```
offset {0|30|50}
```

Selects the start-of-encryption offset (in bytes) into the packet after the MACsec header. Default 0 bytes.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Enabling confidentiality with an offset of 30 bytes:

```
switch(config-macsec-policy)# confidentiality offset 30
```

Disabling confidentiality

```
switch(config-macsec-policy)# no confidentiality
```

## include-sci-tag

### Syntax

```
include-sci-tag  
no include-sci-tag
```

## Description

Within the MACsec policy context, enables inclusion of the Secure Channel Identifier (SCI) tag in the Security TAG (SecTAG) field of the MACsec header. This is the default.

Inclusion of the SCI tag is not required on point-to-point links if the transmitting link has only one MACsec peer.



---

On the 8360 Switch Series model JL717A, inclusion (or exclusion) of the SCI tag must be set identically at both ends of a MACsec channel. Asymmetric SCI tag settings are not supported.

---

The `no` form of this command disables inclusion of the Secure Channel Identifier (SCI) tag in the Security TAG (SecTAG) field of the MACsec header.

## Command context

`config-macsec-policy`

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Enabling the SCI tag:

```
switch(config-macsec-policy) # include-sci-tag
```

Disabling the SCI tag:

```
switch(config-macsec-policy) # no include-sci-tag
```

## macsec policy

### Syntax

```
macsec policy <MACSEC-POLICY-NAME>  
no macsec policy <MACSEC-POLICY-NAME>
```

### Description

Creates the specified MACsec policy and then enters its context (displayed in the CLI as `config-macsec-policy`). If the MACsec policy already exists, this command enters the specified MACsec policy context.

A MACsec policy can be applied to one or more switch ports, enabling MACsec on the ports. An MKA (MACsec Key Agreement) policy must be applied to the same ports.

The `no` form of this command deletes the MACsec policy.



---

A MACsec policy cannot be deleted if it is currently applied to any ports. All application of the policy must be removed before the policy can be deleted.

---

## Command context

`config`

## Parameters

<MACSEC-POLICY-NAME>

Specifies the MACsec policy name. Range: 1 to 32 alphanumeric characters including only the three special characters "." (period), "-" (hyphen), and "\_" (underscore).

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Creating a MACsec policy:

```
switch(config)# macsec policy MS_Policy1
switch(config-macsec-policy)#
```

Deleting a MACsec policy (the policy cannot be currently applied to any ports):

```
switch(config)# no macsec policy MS_Policy1
```

# replay-protection

## Syntax

```
replay-protection [window-size <WINDOW-SIZE>]
no replay-protection
```

## Description

Within the MACsec policy context, enables replay protection with the default or specified window size. With replay protection enabled, packets are expected to arrive within the replay protection window number of packets. For example with a window size of 10, any packet arriving out-of-sequence by more than 10 packets will be discarded. A window size of 0 (the default) enforces strict order of packet reception, discarding all packets not received in perfect sequence.

The `no` form of this command disables replay protections and resets the window size to its 0 default.

## Command context

config-macsec-policy

## Parameters

<WINDOW-SIZE>

Specifies the replay protection window size in packets. Default 0 packets. Range: 0 to 4294967295 packets.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Enabling replay protection with the default window size of 0 (strict order of packet reception):

```
switch(config-macsec-policy)# replay-protection
```

Enabling replay protection with a windows size of 100 packets:

```
switch(config-macsec-policy)# replay-protection window-size 100
```

Disabling replay protection.

```
switch(config-macsec-policy)# no replay-protection
```

## show macsec policy

### Syntax

```
show macsec policy [<MACSEC-POLICY-NAME>]
```

### Description

Shows information for one or all MACsec policies.

### Command context

Operator (>) or Manager (#)

### Parameters

*<MACSEC-POLICY-NAME>*

Specifies an existing MACsec policy name. Range: 1 to 32 alphanumeric characters including only the three special characters "." (period), "-" (hyphen), and "\_" (underscore).

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Examples

Showing information for a specific MACsec policy:

```
switch# show macsec policy MS_Policy1

MACsec Policy Details

Policy Name: MS_Policy1
-----
Cipher suite           : GCM-AES-128
Include SCI            : Yes
Confidentiality        : Enabled
Confidentiality offset : 0
Replay protection      : Enabled
Replay protection window : 0
```

## show macsec statistics

### Syntax

```
show macsec statistics [interface <IF-RANGE>]
```

### Description

Shows MACsec statistics for all MACsec-enabled interfaces or a specific interface or interface range.



## Command context

Operator (>) or Manager (#)

## Parameters

interface <IF-RANGE>

Specifies one or more interfaces for which MACsec statistics information is to be shown.

## Authority

Operators or Administrators or local user group members with execution rights for this command.

Operators can execute this command from the operator context (>) only.

## Examples

Showing MACsec statistics for a specific interface:

```
switch# show macsec statistics 1/1/1

MACsec Statistics

Interface 1/1/1
=====

Rx Statistics
-----
  Unicast Uncontrolled Packets      : 0
  Multicast Uncontrolled Packets    : 604441
  Broadcast Uncontrolled Packets    : 12
  Rx Uncontrolled Drop Packets      : 0
  Rx Uncontrolled Error Packets     : 0
  Rx Controlled Unicast Packets     : 0
  Rx Controlled Multicast Packets   : 2
  Rx Controlled Broadcast Packets   : 12
  Rx Controlled Drop Packets        : 0
  Rx Controlled Error Packets       : 0
  Uncontrolled Octets               : 70196985
  Controlled Octets                 : 6466885

Tx Statistics
-----
  Unicast Uncontrolled Packets      : 0
  Multicast Uncontrolled Packets    : 744632
  Broadcast Uncontrolled Packets    : 16
  Rx Uncontrolled Drop Packets      : 0
  Rx Uncontrolled Error Packets     : 0
  Unicast Controlled Packets        : 0
  Multicast Controlled Packets      : 448486
  Broadcast Controlled Packets      : 125916
  Rx Controlled Drop Packets        : 0
  Rx Controlled Error Packets       : 0
  Uncontrolled Octets               : 98851304
  Controlled Octets                 : 52810544
  Common Octets                     : 169728074

SecY Statistics
-----
  Port Identifier : 1

  Rx Statistics
  -----
```

```
Transform Error Packets : 0
Control Packets         : 3
Untagged Packets        : 0
No Tag Packets          : 0
Bad Tag Packets         : 0
No SCI Packets          : 0
Unknown SCI Packets     : 0
Tagged Control Packets  : 0
Overrun Packets         : 0
```

#### Tx Statistics

```
-----
Transform Error Packets : 0
Control Packets         : 0
Untagged Packets        : 0
```

#### Transmit Secure Channel

```
-----
SCI   : 000C29F6A438004C
```

#### Statistics

```
-----
Encrypted Packets : 35531523571
Protected Packets : 27631523571
```

#### Secure Association

```
-----
Association Number : 0 (old)
```

#### Statistics

```
-----
Encrypted Packets      : 35531523571
Encrypted Octets       : 2204584937326
Protected Packets      : 27631523571
Protected Octets       : 1507619261490
Too Long Packets       : 0
SA Not In Use Packets  : 0
```

#### Receive Secure Channel

```
-----
SCI   : 000C29F6A438003B
```

#### Statistics

```
-----
Late Packets          : 0
Not Valid Packets     : 0
Delayed Packets       : 0
Ok Packets            : 0
```

#### Secure Association

```
-----
Association Number : 0 (old)
```

#### Statistics

```
-----
Unchecked Packets      : 0
Delayed Packets        : 0
Late Packets           : 0
Ok Packets             : 35531534170
Invalid Packets        : 0
Not Valid Packets      : 0
Not Using SA Packets   : 0
```

```
Unused SA Packets      : 0
Decrypted Octets       : 2204607911130
Validated Octets       : 0
```

## show macsec status

### Syntax

```
show macsec status [interface <IF-RANGE>] [detailed]
```

### Description

Shows MACsec status information for all MACsec-enabled interfaces or a specific interface or interface range.

### Command context

Operator (>) or Manager (#)

### Parameters

interface <IF-RANGE>

Specifies one or more interfaces for which MACsec status information is to be shown.

detailed

Specifies that detailed status information is to be shown.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Usage

Applicable to when the `detailed` parameter is included: The stop time for the MACsec secure channel and secure association is updated only when the secure channel or association entry is being deleted. Therefore, it is never shown as set in the `show macsec status detailed` command output.

### Examples

Showing MACsec summary information for all interfaces:

```
switch# show macsec status

MACsec Protocol Status

Interface  Port ID  Policy                Protection          Status  State
-----
1/1/1      0         MS_Policy1           Conf, Offset 0     Up      Retire
1/1/2      0         MS_Policy1           IC                  Down    Init
...
```

Showing detailed MACsec information for a specific interface:

```
switch# show macsec status interface 1/1/1 detailed

Interface 1/1/1
=====
```

```

Port Identifier: 0
=====

Policy          : MS_Policy1
Status          : Up
State           : Retire
Cipher Suite    : GCM-AES-128
Protection      : Conf, Offset 0

Transmit Secure Channel
-----
SCI   : 000C29F6A4380004C
SSCI  : 1

Secure Association
-----
Association Number : 0 (old)
Key Identifier     : 4F18CE25228178FD15976E4C
Packet Number     : 9500
SA-Start-Time      : Sun Oct 18 04:05:11 UTC 2020
SA-Stop-Time       : Sun Oct 18 04:10:12 UTC 2020

Association Number : 1 (current)
Key Identifier     : 4F18CE25228178FD15976E4C
Packet Number     : 19000
SA-Start-Time      : Sun Oct 18 04:10:13 UTC 2020
SA-Stop-Time       : -

Receive Secure Channel
-----
SCI   : 000C29F6A4360003B
SSCI  : 2

Secure Association
-----
Association Number : 0 (old)
Key Identifier     : 4F18CE25228178FD15976E4C
Lowest Packet Number : 9500
SA-Start-Time      : Sun Oct 18 04:05:12 UTC 2020
SA-Stop-Time       : Sun Oct 18 04:10:12 UTC 2020

Association Number : 1 (current)
Key Identifier     : 4F18CE25228178FD15976E4C
Lowest Packet Number : 19000
SA-Start-Time      : Sun Oct 18 04:10:13 UTC 2020
SA-Stop-Time       : -

```

## MKA commands (MACsec)




---

MACsec is exclusive to the 8360 Switch Series, model JL717A.

---

### apply mka policy

#### Syntax

```

apply mka policy <MKA-POLICY-NAME>
no apply mka policy

```

#### Description

Within the selected interface context, applies the specified MKA policy to the selected port. To start the MKA protocol on the port, a MACsec policy must also be applied to the port.



---

An MKA policy can be applied to a physical interface port that is not part of any LAG ports or to a lag port. It can also be applied to an interface that is configured as an MCLAG, VSX keep-alive, or VSX inter-switch-link.

---

If an MKA policy is already applied to the selected port, this command replaces the existing policy application.

The `no` form of this command dissociates the specified policy from the port.

## Command context

`config-if`

## Parameters

`<MKA-POLICY-NAME>`

Specifies the MKA policy name. Range: 1 to 32 alphanumeric characters including only the three special characters "." (period), "-" (hyphen), and "\_" (underscore).

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

- When any MACsec or MKA policy parameter is updated, any active MACsec session on all interfaces running the MACsec or MKA policy is terminated and restarted. This is indicated with the following prompt that provides an opportunity to not execute the `apply` command.

```
This policy is currently in use by one or more interfaces.  
Updating the policy will cause existing MACsec sessions using  
the policy to restart.  
Continue (y/n)?
```

- For non-LAG ports, a range of ports can be specified in the `interface` command used to enter the interface context. For example, entering the interface context for ports 1/1/1 through 1/1/4:

```
switch(config)# interface 1/1/1-1/1/4  
switch(config-if-<1/1/1-1/1/4>)# apply mka policy MKA_Policy1
```

- Not all interfaces on a switch may support the MACsec capability. An error will be generated when a policy is applied to a physical interface that is not capable of MACsec. For LAG ports, any non-MACsec capable interfaces that are part of the LAG will be blocked.

## Examples

Applying an MKA policy to a range of two ports:

```
switch(config)# interface 1/1/1-1/1/2  
switch(config-if-<1/1/1-1/1/2>)# apply mka policy MKA_Policy1
```

Attempting to apply an MKA policy to a port that is not MACsec capable:

```
switch(config)# interface 1/1/5
switch(config-if)# apply mka policy MKA_Policy1

MACsec is not supported on the interface.
switch(config-if)#
```

Removing MKA policy association from a port:

```
switch(config)# interface 1/1/1
switch(config-if)# no apply mka policy
```

Applying an MKA policy to a LAG port:

```
switch(config)# interface lag 1
switch(config-if)# apply mka policy MKA_Policy1
```

## clear mka statistics

### Syntax

```
clear mka statistics [interface <IF-RANGE>]
```

### Description

Clears MKA statistics on all MACsec-enabled interfaces or on a specific interface or interface range. MKA statistics are cleared for the entire switch rather than just in the current user session.

### Command context

Operator (>) or Manager (#)

### Parameters

```
interface <IF-RANGE>
```

Specifies one or more interfaces (ports) for which MKA statistics information is to be cleared.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Examples

Clearing MKA statistics on an interface range:

```
switch# clear mka statistics 1/1/1-1/1/4
```

Clearing MKA statistics on all MACsec-enabled interfaces:

```
switch# clear mka statistics
```

## key-server-priority

### Syntax

```
key-server-priority <PRIORITY>
no key-server-priority
```

## Description

In the `config-mka-policy` policy context, configures the MKA key server priority. The highest priority is 0 and indicates that this switch strongly wants to be the MKA key server. The lowest priority is 255 and indicates that switch does not want to be the MKA key server, allowing the switch at the other end of the link to be the key server. Set this priority on the switches at either end of the link to achieve the desired effect.

If the key server priority is 0 on both switches then the switch with the lowest system MACsec address is elected as key server.

The `no` form of this command resets the MKA key server priority to its default of 0.

## Command context

```
config-mka-policy
```

## Parameters

<PRIORITY>

Selects the MKA key server priority for this switch. Default 0 (highest priority). Range: 0 to 255.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Setting the MKA key server priority:

```
switch(config-mka-policy) # key-server-priority 5
```

Resetting the MKA key server priority to its default of 0:

```
switch(config-mka-policy) # no key-server-priority
```

# mka policy

## Syntax

```
mka policy <MKA-POLICY-NAME>
no mka policy <MKA-POLICY-NAME>
```

## Description

Creates the specified MKA (MACsec Key Agreement) policy and then enters its context (displayed in the CLI as `config-mka-policy`). If the MKA policy already exists, this command enters the specified MKA policy context.

An MKA policy can be applied to one or more switch ports, enabling MKA on the ports. A MACsec policy must be applied to the same ports.

The `no` form of this command deletes the MKA policy.



---

An MKA policy cannot be deleted if it is currently applied to any ports. All application of the policy must be removed before the policy can be deleted.

---

## Command context

config

## Parameters

*<MKA-POLICY-NAME>*

Specifies the MKA policy name. Range: 1 to 32 alphanumeric characters including only the three special characters "." (period), "-" (hyphen), and "\_" (underscore).

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Creating an MKA policy:

```
switch(config)# mka policy MKA_Policy1
switch(config-mka-policy)#
```

Deleting an MKA policy (the policy cannot be currently applied to any ports):

```
switch(config)# no mka policy MKA_Policy1
```

# pre-shared-key

## Syntax

```
pre-shared-key ckn <CA-KEY-NAME> cak {plaintext [<PLAINTEXT-CAK>] | ciphertext <CIPHERTEXT-CAK>}
no pre-shared-key
```

## Description

In the `config-mka-policy` policy context, configures the pre-shared key by setting the CKN (Connectivity Association Key Name) and the CAK (Connectivity Association Key). When plaintext CAK is specified without providing the CAK, it is prompted.

The `no` form of this command deletes the PSK configuration including the CKN and CAK.

## Command context

config-mka-policy

## Parameters

*<CA-KEY-NAME>*

Specifies the CKN (Connectivity Association Key Name). Range: 1 to 64 hexadecimal characters.

plaintext [<PLAINTEXT-CAK>]

Specifies the CAK (Connectivity Association Key) in plaintext. Range: 1 to 64 hexadecimal characters.

ciphertext <CIPHERTEXT-CAK>

Specifies the CAK (Connectivity Association Key) as ciphertext.



## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring the pre-shared key with a specified plaintext CAK:

```
switch(config-mka-policy) # pre-shared-key ckn abcdef12 cak plaintext 123abcdef
```

Configuring the pre-shared key with a prompted plaintext CAK:

```
switch(config-mka-policy) # pre-shared-key ckn abcdef12 cak plaintext
Enter CAK: *****
Confirm CAK: *****
```

Configuring the pre-shared key with a ciphertext CAK:

```
switch(config-mka-policy) # pre-shared-key ckn abcdef12 cak ciphertext
AQBApUvjDZgUxtTpgA4NLqnsn7CjXqbDch+BOS7y9fcWExLUBgAAAKUmDYdhew==
```

Deleting the PSK configuration including its CKN and CAK:

```
switch(config-mka-policy) # no pre-shared-key
```

## show mka policy

### Syntax

```
show mka policy [<MKA-POLICY-NAME>]
```

### Description

Shows information for one or all MKA policies.

### Command context

Operator (>) or Manager (#)

### Parameters

<MKA-POLICY-NAME>

Specifies an existing MKA policy name. Range: 1 to 32 alphanumeric characters including only the three special characters "." (period), "-" (hyphen), and "\_" (underscore).

## Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

## Examples

Showing information for a specific MKA policy:

```
switch# show mka policy MKA_Policy1

MKA Policy Details

Policy Name: MKA_Policy1
-----
Mode                : Pre-shared key
CKN                 : abcdef12
CAK (encrypted)     :
AQBapvjDZUxtTgA4Lqnsn7CjqbDch+BS7y9fcWExLUBgAAAKUmDYdhew==
Key-server Priority  : 5
Transmit Interval   : 4 seconds
```

## show mka statistics

### Syntax

```
show mka statistics [interface <IF-RANGE>]
```

### Description

Shows MKA statistics for all MACsec-enabled interfaces or a specific interface or interface range. The MKA statistics are refreshed periodically, approximately every five seconds.

### Command context

Operator (>) or Manager (#)

### Parameters

```
interface <IF-RANGE>
```

Specifies one or more interfaces for which MKA statistics are to be shown.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Examples

Showing MKA statistics information for a specific interface:

```
switch# show mka statistics 1/1/1

MKA Statistics

Interface 1/1/1
=====

KaY
----
SCI : 000C29F6A4380004C

Statistics
-----
MKPDUs With Invalid Version : 0
MKPDUs With Invalid CKN     : 0

Participant
```

```

-----
CKN : ABCDEF12

Statistics
-----
Tx MKPDUs           : 16534893
Rx MKPDUs           : 16534893
SAKs Distributed     : 0
SAKs Received        : 0
MKPDUs With Invalid ICV : 0
MKPDUs With Duplicate MI : 0
MKPDUs With Invalid MN : 0
...

```

## show mka status

### Syntax

```
show mka status [interface <IF-RANGE>]
```

### Description

Shows MKA status information for all MACsec-enabled interfaces or a specific interface or interface range.

### Command context

Operator (>) or Manager (#)

### Parameters

```
interface <IF-RANGE>
```

Specifies one or more interfaces for which MKA status information is to be shown.

### Authority

Operators or Administrators or local user group members with execution rights for this command. Operators can execute this command from the operator context (>) only.

### Examples

Showing MKA status information for a specific interface:

```

switch# show mka status 1/1/1

Interface 1/1/1
=====

MKA Port Identifier : 1
MKA Session Status  : Secured
Mode                 : Pre-shared key
CKN                  : abcdef12
CAK (encrypted)      : AQBapvjDZUxtTgA4Lqnsn7CjqbDch+BS7y9fcWExLUBgAAAKUmDYdhew==
Member Identifier    : 1c64f054f894b5482defdf81
Message Number       : 86
Capability            : Conf, Offset 0
Transmit Interval    : 4 seconds
Key Server Priority   : 5
Key Server           : No

Live Peer List:

```

MI	MN	PRI	Capability	Rx-SCI
fb7f82788e4cd38dbc65dc55	119	16	IC, Conf, Offset 0	a45d36489bfe0002

Potential Peer List:

MI	MN	PRI	Capability	Rx-SCI
...				

## transmit-interval

### Syntax

```
transmit-interval <INTERVAL>
no transmit-interval
```

### Description

In the `config-mka-policy` policy context, configures the MKA packet transmit interval. The `no` form of this command resets the MKA packet transmit interval to its default of 2 seconds.

### Command context

`config-mka-policy`

### Parameters

<INTERVAL>

Selects the MKA packet transmit interval. Default 2 seconds. Range: 2 to 6 seconds.

### Authority

Administrators or local user group members with execution rights for this command.

### Examples

Setting the MKA packet transmit interval:

```
switch(config-mka-policy)# transmit-interval 4
```

Resetting the MKA packet transmit interval to its default of 2 seconds:

```
switch(config-mka-policy)# no transmit-interval
```

Several measures can be taken to enhance switch security, including setting secure mode to enhanced in the Service OS. For maximum security, perform all the configuration described in this chapter.

## Configuring enhanced security

### Prerequisites

If you have switch configuration that you want to retain, create a backup. This procedure erases all configuration, including the current running configuration, the startup configuration, and all historical configuration checkpoints.

### Procedure

1. Set enhanced security mode:
  - a. Reboot the switch into the Service OS with command `boot system serviceos`. If on an 8400 Switch with both Management Modules:
    - i. Issue the `boot` command only on the active Management Module. This command ensures that both Management Modules are booted into the Service OS.
    - ii. Perform steps b to e on both modules starting with the active module.
  - b. Log in to the Service OS as `admin`.
  - c. Enter command `secure-mode enhanced`.
  - d. When prompted about the mode change, respond with `y` for "yes."
  - e. Wait for the reboot and zeroization to complete. The switch firmware boots automatically.
2. Ensure adequate password requirements:
  - a. Before adding users, enable and configure password complexity as described in [password complexity](#). To maintain enhanced security, configure the `password complexity` subcommand settings no smaller than their defaults.
  - b. Configure passwords for all users, including `admin`. To make your password complexity settings applicable to the default admin user, change the admin password after enabling password complexity. The new admin password must respect your password complexity settings.
3. Ensure proper login management as follows:
  - a. Configure local user session management as described in [CLI user session management commands](#) using `cli-session` and its subcommands `max-per-user`, `timeout`, and `tracking-range` to achieve the wanted configuration. To maintain enhanced security, configure `cli-session` subcommand settings no smaller than their defaults.
  - b. Restrict remote SSH connections to only use certified crypto algorithms using `ssh certified-algorithms-only`.
  - c. Configure pre- and post-login banners using respectively, `banner motd`, and `banner exec`.

4. Ensure that the switch date and time is accurately set using `clock datetime <DATE> <TIME>`.
5. When logging to a remote syslog server is required, ensure that the connection to the server is cryptographically secure. See [Configuring remote logging using SSH reverse tunnel](#).

To ensure that enhanced security is maintained, also respect these requirements:

- Do not configure remote logging with a remote server directly without setting up an SSH tunnel.
- Do not configure passwords and secret keys using the plaintext option.



---

When in enhanced security mode, the switch (Product OS) `start-shell` command is disabled for security purpose. If you attempt to use this command while in enhanced security mode, it is rejected and the following error message is displayed:

```
The start-shell command is not available in enhanced secure mode.
```

---



---

When in enhanced security mode, the following Service OS commands are disabled for security purposes: `config-clear`, `password`, `sh`, and `update`. If you attempt to use any of these Service OS commands while in enhanced security mode, the command is rejected and an error message is displayed.

---

## password complexity

### Syntax

```
password complexity
no password complexity
```

### Description

Enters the password-complexity context (shown in the switch prompt as `config-pwd-cplx`) for the purpose of enabling and configuring password complexity. Password complexity enhances security by enforcing specific password complexity requirements. Password complexity is disabled by default and must be enabled by execution of the `enable` command.

The no form of this command reverts all settings to their default values and disables password complexity enforcement.



---

To ensure that enhanced security is maintained, it is recommended that you do not set any values to less than their defaults.

---



---

Password complexity applies only to local authentication. For remote authentication, you may choose to set up an equivalent of password complexity according to whatever is supported on your particular TACACS+ or RADIUS server.

---

### Command context

```
config
```

### Subcommands

These subcommands are available within the password complexity context (shown in the switch prompt as `config-pwd-cplx`).

```
enable
```

Enables password complexity enforcement. The enforcement only applies to passwords created after this enabling. Existing passwords are not checked against password complexity.

`disable`

Disables password complexity enforcement.

`[no] history-count <COUNT>`

Specifies the number of previous passwords checked to prevent excessive reuse. Not applicable when adding new users. The no form of this subcommand resets the value to its default. Default: 5. Range: 1 to 5.

`[no] minimum-length <LENGTH>`

Specifies the minimum password length. The no form of this subcommand resets the value to its default. Default: 8. Range: 1 to 32.

`[no] position-changes <POSITIONS>`

Specifies the minimum number of characters that must change in the new password compared to the previous password. Not applicable if no previous password exists, including when adding new users. The no form of this subcommand resets the value to its default. Default: 8. Range: 1 to 32.

The number of password position changes is based on the number of simple character insertions, deletions, or replacements. For example:

Old password: abCD4\$ New password: abCD\$ Position changes=1 ("4" deleted) Old password: abCD4\$ New password: abCDEF4\$ Position changes=2 ("EF" inserted) Old password: abCD4\$ New password: ebCD4\$1 Position changes=2 ("a" replaced with "e," "1" added) Old password: abCD4\$ New password: abC\$# Position changes=3 ("D4" deleted, "#" added)

`[no] lowercase-count <COUNT>`

Specifies the minimum lowercase character count for new passwords. The no form of this subcommand resets the value to its default. Default: 1. Range: 0 to 32.

`[no] uppercase-count <COUNT>`

Specifies the minimum uppercase character count for new passwords. The no form of this subcommand resets the value to its default. Default: 1. Range: 0 to 32.

`[no] numeric-count <COUNT>`

Specifies the minimum numeric digit count for new passwords. The no form of this subcommand resets the value to its default. Default: 1. Range: 0 to 32.

`[no] special-char-count <COUNT>`

Specifies the minimum special character count for new passwords. The no form of this subcommand resets the value to its default. Default: 1. Range: 0 to 32.

`list`

List the subcommands available within the password complexity context.

`exit`

Exits the password complexity context.

`end`

Exits the password complexity context and then the config context.

## Authority

Administrators or local user group members with execution rights for this command.

## Usage

- Password complexity is only for use with plaintext passwords. With password complexity enabled, existing ciphertext passwords will continue working until a password is changed. All new passwords must be entered in plaintext form and be compliant with your password complexity configuration.
- The effective minimum password length may be larger than the configured `minimum-length` value. The effective minimum password length is calculated as follows:

LARGEST-of: (minimum-length, position-changes, (SUM-of: lowercase-count+uppercase-count+numeric-count+special-char-count))

For example, with minimum-length=8, and position-changes=10 (and the sum of the other four count settings <=9), the **effective minimum-length is 10** (because position-changes is largest). Similarly, with a minimum-length=12, position-changes=8, lowercase-count=8, uppercase-count=4, numeric-count=1, special-char-count=1, the **effective minimum-length is 14** (8+4+1+1=14) (because sum off the four counts is largest).

## Examples

Configuring password complexity settings with an effective minimum length of 10 (because position-changes is 10):

```
switch(config)# password complexity
switch(config-pwd-cplx)# history-count 3
switch(config-pwd-cplx)# minimum-length 8
switch(config-pwd-cplx)# position-changes 10
switch(config-pwd-cplx)# lowercase-count 2
switch(config-pwd-cplx)# uppercase-count 2
switch(config-pwd-cplx)# numeric-count 2
switch(config-pwd-cplx)# special-char-count 2
switch(config-pwd-cplx)# enable
switch# exit
```

Configuring password complexity settings with an effective minimum length of 14 (because the sum of the four count items is 14):

```
switch(config)# password complexity
switch(config-pwd-cplx)# history-count 4
switch(config-pwd-cplx)# minimum-length 12
switch(config-pwd-cplx)# position-changes 8
switch(config-pwd-cplx)# lowercase-count 8
switch(config-pwd-cplx)# uppercase-count 4
switch(config-pwd-cplx)# numeric-count 1
switch(config-pwd-cplx)# special-char-count 1
switch(config-pwd-cplx)# enable
switch# exit
```

Enabling password complexity (with default settings) and changing a user (admin1) password successfully but failing to change another user (admin2) password due to not meeting complexity requirements:

```
switch(config)# password complexity
switch(config-pwd-cplx)# enable
switch(config-pwd-cplx)# exit
switch(config)#
switch(config)# user admin1 password
Changing password for user admin1
Enter old password:*****
Enter new password:*****
Confirm new password:*****
switch(config)#
switch(config)# user admin2 password
Changing password for user admin2
Enter old password:*****
Enter new password:*****
Confirm new password:*****
User password not changed.
```



```
The new password does not meet one or more of the following complexity requirements:
Minimum length      : 8
Position changes    : 8
Numeric count       : 1
Lowercase count     : 1
Uppercase count     : 1
Special character count : 1
switch(config)#
```

With password complexity already enabled, attempting to change an existing user password but failing because the new password is identical to a recently used one (`history-count`).

```
switch(config)# user admin1 password
Changing password for user admin1
Enter old password:*****
Enter new password:*****
Confirm new password:*****
User password not changed.
The new password is the same as a recently used password.
switch(config)#
```

With password complexity already enabled, creating a new admin user (admin3) with a plaintext password that meets complexity requirements.

```
switch(config)# user admin3 group administrators password
Adding user admin3
Enter password:*****
Confirm password:*****
switch(config)#
```

With password complexity already enabled, attempting to create a new admin user (admin4) with a ciphertext password but failing because ciphertext passwords are not supported with password complexity enabled.

```
switch(config)# user admin4 group administrators password ciphertext AQBapPd...==
Ciphertext passwords cannot be used when password complexity is enabled.
switch(config)#
```

## Configuring remote logging using SSH reverse tunnel

Logging to a remote syslog server can be made cryptographically secure by using SSH reverse tunnel. The `syslog` daemon on the switch forwards log messages to the SSH tunnel, and the SSH tunnel endpoint on the remote server host forwards messages to the listening `syslog` server.



---

This procedure includes sample configuration commands for a user-supplied syslog server based on Ubuntu 14.04.5 LTS with `rsyslog`. It is up to the user to check their server documentation and adjust the sample commands as required. Optionally see your server documentation for information on how to use the `systemd` and `autossh` services to automatically restore the SSH tunnel after system reboot.

---

## Prerequisites

The user-supplied remote syslog server must be on a network that can reach the switch management interface.

## Procedure

1. Configure SSH server on the switch.
  - a. Enter these commands (although this example uses the mgmt VRF, other VRFs can be used):

```
switch(config)# interface mgmt
switch(config-if-mgmt)# no shutdown
switch(config-if-mgmt)# ip address <switch_mgmt_IP>
switch(config-if-mgmt)# exit
switch(config)# ssh server vrf mgmt
```
  - b. If public key authentication is desired for remote SSH users, configure it on the switch:

```
switch(config)# user admin authorized-key <PUBKEY>
```
2. Configure logging on the switch to forward to localhost:

```
switch(config)# logging localhost tcp <switch_tcp_port> vrf mgmtinclude-auditable-events
```
3. Configure the `rsyslog` server on the remote host:
  - a. Make `rsyslog` accept TCP connections and specify the log file, by adding the following to `/etc/rsyslog.conf`:

```
$ModLoad imtcp
$InputTCPServerRun <server_tcp_port>
$template RemoteLogs,"/var/log/remote.log"
*. * ?RemoteLogs
```
  - b. To activate the added configuration, restart the `rsyslog` server:

```
root@Ubuntu4479:~#sudo service rsyslog restart
```
4. Establish an SSH reverse tunnel from the remote host to the switch:

```
root@Ubuntu4479:~#ssh -nNTx -R
<switch_tcp_port>:127.0.0.1:<server_tcp_port>
admin@<switch_mgmt_IP>
```

## CLI user session management commands

### cli-session

#### Syntax

```
cli-session
no cli-session
```

#### Description

Enters the CLI session context (shown in the switch prompt as `config-cli-session`) for the purpose of configuring CLI user session management. Session management enhances security by enforcing specific CLI user session requirements. The following information is provided at time of successful login:

- When applicable, the number of failed login attempts since the most recent successful login.
- The date, time, and location (console or IP address or hostname) of the most recent previous successful login.
- The count of successful logins within the past (configurable) time period.

For example:

```
switch login: admin
Password:
```

There were 3 failed login attempts since the last successful login  
Last login: 2019-04-20 08:51:33 from the console  
User "admin" has logged in 73 times in the past 30 days

The no form of this command disables concurrent CLI user session restrictions and reverts `timeout` and `tracking-range` to their default values.



---

To ensure that enhanced security is maintained, it is recommended that you keep CLI user session management fully enabled by setting `max-per-user` to a nondefault value.

---



---

The `cli-session` command applies only to SSH/console login connection types. It does not apply to other connection types such as REST.

---

## Command context

config

## Subcommands

These subcommands are available within the CLI session context.

[no] `max-per-user` *<SESSIONS>*

Specifies the maximum number of concurrent CLI sessions per user. The no form of this subcommand disables concurrent CLI user session restrictions. Default: Disabled (no value). Range: 1 to 5.



---

When the same user name is configured for both local and remote authentication, both users, regardless of privilege level, are considered to be the same user for the purpose of counting concurrent CLI sessions. For example, with `max-per-user` set to 1 and user `admin1` configured for local and remote authentication, only the local user `admin1` or the remote user `admin1` can be logged in at any given moment. Both `admin1` users cannot be logged in simultaneously unless `max-per-user` is increased to at least 2.

---

[no] `timeout` *<MINUTES>*

Specifies the number of minutes a CLI session can be idle before the session is automatically terminated and the user is logged out. A value of 0 minutes disables the session timeout. The no form of this subcommand sets the timeout value to the default. Default 30: Range 0 to 4320.



---

This subcommand is the recommended replacement for the `session-timeout` command.

---

[no] `tracking-range` *<DAYS>*

Specifies the maximum number of days to track CLI user session logins. The no form of this subcommand resets the value to its default. Default 30: Range 1 to 30.

`exit`

Exits the CLI session context.

`end`

Exits the CLI session context and then the config context.

## Authority

Administrators or local user group members with execution rights for this command.

## Examples

Configuring CLI user session settings for a maximum of one concurrent session, a 20-minute timeout, and tracking for a maximum of 25 days.

```
switch(config)# cli-session  
switch(config-cli-session)# max-per-user 1  
switch(config-cli-session)# timeout 20  
switch(config-cli-session)# tracking-range 25  
switch# exit
```

After successful earlier logins, logging in from the console without any intervening unsuccessful logins.

```
switch login: admin1  
Password:  
  
Last login: 2019-04-15 14:10:21 from the console  
User 'admin1' has logged in 65 times in the past 25 days
```

Attempting to log in as `admin1` when already logged in as `admin1` from elsewhere.

```
switch login: admin1  
Password:  
Too many logins for 'admin1'
```

After successful earlier logins, attempting to log in twice with an invalid password, followed by a successful login.

```
switch login: admin1  
Password:  
  
Login incorrect  
switch login: admin1  
Password:  
  
Login incorrect  
switch login: admin1  
Password:  
  
There were 2 failed login attempts since the last successful login  
Last login: 2019-04-15 17:22:45 from 192.168.1.1  
User 'admin1' has logged in 72 times in the past 25 days
```



---

Applies to the Aruba 8400 Switch Series only, which must be running AOS-CX 10.07.0030 or later.

---

AOS-CX switches consist of automatic detection and control for certain link errors and excessive traffic conditions. Fault monitor can be used to log an event or send SNMP traps for these conditions and temporarily disable the port to protect the network. Monitoring can be enabled for all recognized faults or for individual faults after the threshold action and auto-enable parameters for each can be configured. Fault monitor applies only to physical ports and not to LAGs, tunnels, VSF links, or other types of interfaces. Faults can be applied to the individual members of a LAG.

## Fault monitoring conditions

The following fault conditions are monitored:

### Excessive oversize packets

An excessive oversized packet fault is reported when the amount of ingress oversized frames per 10,000 received frames exceeds the configured threshold value in a twenty second interval. In oversize packet fault, the packets size is more than the configured MTU on the interface with good cyclic redundancy check (CRC).

### Excessive fragments

An excessive fragment fault is reported when the amount of ingress fragment frames per 10,000 received frames exceeds the configured threshold value in a twenty second interval. In fragments fault, the packet size is lesser than the configured MTU on the interface with bad CRC.

### Excessive CRC errors

An excessive CRC error fault is reported when the amount of ingress `crc-error` frames per 10,000 received frames exceeds the configured threshold value in a twenty second interval.

### Excessive TX drops

Excessive TX drops fault is an example of over bandwidth. It is reported when egress dropped packets per 10,000 transmitted frames exceeds the configured threshold value in a twenty second interval.

### Excessive link flaps

A link flap fault is reported when the count of transitions between link-up and link-down state exceeds the configured threshold in a ten second interval.

### Excessive broadcasts

A broadcast storm fault is reported when the average ingress traffic rate of broadcast packets exceeds the configured threshold in a twenty second interval.

The default threshold level is configured as a percentage of the bandwidth of the port. Larger the frame size, smaller the converted threshold value in PPS. Hence larger frames require lower threshold percent configurations to hit the fault.

## Excessive multicasts

A multicast storm fault is reported when the average ingress traffic rate of multicast packets exceeds the configured threshold in a twenty second interval.

## Excessive collisions

An excessive collision fault is an example of over bandwidth, it gets reported when egress `collision` frames per 10,000 transmitted frames exceeds the configured threshold value in a 20 second interval.

## Excessive Late Collisions

An excessive late collision fault is reported when ingress `late-collision` frames per 10,000 received frames exceeds the configured threshold value in a 300 second interval.

## Excessive alignment errors

A full duplex mismatch fault is reported when ingress `alignment-error` frames per 10,000 received frames exceeds the configured threshold value in a 20 second interval.

## Fault monitor commands

### (enabling, disabling faults)

```
{all | <FAULT>}  
no {all | <FAULT>}
```

### Description

Within the selected fault monitor profile context, enables all faults or specific faults for monitoring.



Faults enabled with this command use default actions and thresholds unless the actions and thresholds are configured with respective commands `action` and `threshold`.

By default, all faults are disabled in a profile and remain disabled until enabled as described here. Configuring the action and threshold does not enable the fault.

The `no` form of this command disables faults for monitoring.

Parameter	Description
<code>all</code>	Selects all faults.
<code>&lt;FAULT&gt;</code>	Selects a specific fault. Available fault names: <code>excessive-crc-errors</code> <code>excessive-oversize-packets</code> <code>excessive-fragments</code> <code>excessive-tx-drops</code> <code>excessive-collisions</code>

Parameter	Description
	excessive-late-collisions excessive-alignment-errors excessive-link-flaps excessive-broadcasts excessive-multicasts

## Examples

Enabling faults:

```
switch(config-fault-monitor-profile) # all
switch(config-fault-monitor-profile) # excessive-oversize-packets
switch(config-fault-monitor-profile) # excessive-fragments
switch(config-fault-monitor-profile) # excessive-crc-errors
switch(config-fault-monitor-profile) # excessive-tx-drops
switch(config-fault-monitor-profile) # excessive-link-flaps
switch(config-fault-monitor-profile) # excessive-broadcasts
switch(config-fault-monitor-profile) # excessive-multicasts
switch(config-fault-monitor-profile) # excessive-collisions
switch(config-fault-monitor-profile) # excessive-late-collisions
switch(config-fault-monitor-profile) # excessive-alignment-errors
```

Disabling faults:

```
switch(config-fault-monitor-profile) # no all
switch(config-fault-monitor-profile) # no excessive-oversize-packets
switch(config-fault-monitor-profile) # no excessive-fragments
switch(config-fault-monitor-profile) # no excessive-crc-errors
switch(config-fault-monitor-profile) # no excessive-tx-drops
switch(config-fault-monitor-profile) # no excessive-link-flaps
switch(config-fault-monitor-profile) # no excessive-broadcasts
switch(config-fault-monitor-profile) # no excessive-multicasts
switch(config-fault-monitor-profile) # no excessive-collisions
switch(config-fault-monitor-profile) # no excessive-late-collisions
switch(config-fault-monitor-profile) # no excessive-alignment-errors
```

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	config-fault-monitor-profile	Administrators or local user group members with execution rights for this command.

## action

```
{all | <FAULT>} action {notify | notify-and-disable [auto-enable <TIMEOUT>]}
no {all | <FAULT>} action {notify | notify-and-disable [auto-enable <TIMEOUT>]}
```

## Description

Within the selected fault monitor profile context, configures the fault monitoring action for the specified fault. Default action: `notify` with `auto-enable` disabled.

The no form of this command removes the action and disables `auto-enable`.

Parameter	Description
<code>all</code>	Selects all faults.
<code>&lt;FAULT&gt;</code>	Selects a specific fault. Available fault names: <code>excessive-crc-errors</code> <code>excessive-oversize-packets</code> <code>excessive-fragments</code> <code>excessive-tx-drops</code> <code>excessive-collisions</code> <code>excessive-late-collisions</code> <code>excessive-alignment-errors</code> <code>excessive-link-flaps</code> <code>excessive-broadcasts</code> <code>excessive-multicasts</code>
<code>notify</code>	Selects the <code>notify</code> action. Notifies through events, DLOGs, and SNMP trap. This action is enabled by default.
<code>notify-and-disable</code>	Selects the action as <code>notify-and-disable</code> . Notifies through events, DLOGs, and SNMP trap, and then disables the port.
<code>auto-enable &lt;TIMEOUT&gt;</code>	Sets the number of seconds after which a port disabled by the <code>notify-and-disable</code> action is automatically re-enabled. Range: 1 to 604800 seconds.



The fault parameter values are saved even after a fault is disabled in the profile. The saved values will be used if the fault is later re-enabled in the profile again.

## Examples

Configuring the `notify` fault action:

```
switch(config-fault-monitor-profile)# all action notify
switch(config-fault-monitor-profile)# excessive-oversize-packets action notify
switch(config-fault-monitor-profile)# excessive-collisions action notify
```

Configuring the `notify-and-disable` fault action:

```
switch(config-fault-monitor-profile)# all action notify-and-disable
switch(config-fault-monitor-profile)# excessive-oversize-packets action notify-and-disable
switch(config-fault-monitor-profile)# excessive-late-collisions action notify-and-disable
switch(config-fault-monitor-profile)# excessive-alignment-errors action notify-and-disable
```

Configuring the `notify-and-disable` action with `auto-enable`:



```
switch(config-fault-monitor-profile)# excessive-oversize-packets action notify-and-disable auto-enable 80
switch(config-fault-monitor-profile)# excessive-collisions action notify-and-disable auto-enable 70
```

Resetting the fault action to default:

```
switch(config-fault-monitor-profile)# no excessive-oversize-packets action
switch(config-fault-monitor-profile)# no excessive-alignment-errors action
```

Disabling auto-enable:

```
switch(config-fault-monitor-profile)# no all action notify-and-disable auto-enable
switch(config-fault-monitor-profile)# no excessive-alignment-errors action notify-and-disable auto-enable
```

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	config-fault-monitor-profile	Administrators or local user group members with execution rights for this command.

## apply fault-monitor profile

```
apply fault-monitor profile <PROFILE-NAME>
no apply fault-monitor profile [<PROFILE-NAME>]
```

### Description

Applies a fault monitoring profile to the selected interface or interface range.

The **no** form of this command removes the fault monitoring profile from the selected interface or interface range.

Parameter	Description
<PROFILE-NAME>	Specifies the fault monitor profile name. Range: Up to 64 alphanumeric and special characters.

### Examples

Applying the fault monitoring profile to a interface:

```
switch(config)# interface 1/1/1
switch(config-if)# apply fault-monitor profile noisy-ports
```

Applying the fault monitoring profile to a interface range:

```
switch(config)# interface 1/1/2-1/1/24
switch(config-if)# apply fault-monitor profile quiet-ports
```

## Command History

Release	Modification
10.07.0030	Command introduced.

## Command Information

Platforms	Command context	Authority
4100i 8400	config-if	Administrators or local user group members with execution rights for this command.

## fault-monitor profile

```
fault-monitor profile <PROFILE-NAME>
no fault-monitor profile <PROFILE-NAME>
```

## Description

Creates a fault monitoring profile and enters its context. If the profile already exists, this command enters the profile context. A maximum of 16 fault monitoring profiles are supported.

The `no` form of this command deletes the fault monitoring profile.

Parameter	Description
<PROFILE-NAME>	Specifies the fault monitor profile name. Range: Up to 64 alphanumeric and special characters.

## Examples

Creating a fault monitor profile:

```
switch(config)# fault-monitor profile noisy-ports
switch(config-fault-monitor-profile)#
```

Deleting a fault monitor profile:

```
switch(config)# no fault-monitor profile noisy-ports
switch(config)#
```

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	config	Administrators or local user group members with execution rights for this command.

## show fault-monitor profile

show fault-monitor profile <PROFILE-NAME>

### Description

Shows fault monitoring profile information for all profiles or a specific profile.

Parameter	Description
<PROFILE-NAME>	Specifies the fault monitor profile name. Range: Up to 64 alphanumeric and special characters.

### Example

Showing information for all fault monitoring profiles:

```
switch# show fault-monitor profile
-----
Fault monitor profile: noisy-ports
-----
Auto
Fault                               Enabled  Threshold  Action                Enable
-----
excessive-broadcasts                yes      5%          notify-and-disable    --
excessive-multicasts                 yes      1000 pps    notify-and-disable    --
excessive-link-flaps                 yes      7           notify-and-disable    --
excessive-oversize-packets           yes      25          notify-and-disable    --
excessive-fragments                 yes      25          notify-and-disable    --
excessive-crc-errors                 yes      25          notify-and-disable    --
excessive-late-collisions             yes      25          notify-and-disable    --
excessive-collisions                 yes      25          notify-and-disable    --
excessive-tx-drops                   yes      25          notify-and-disable    --
excessive-alignment-errors           yes      25          notify-and-disable    --
-----
Fault monitor profile: quiet-ports
-----
Auto
Fault                               Enabled  Threshold  Action                Enable
-----
excessive-broadcasts                yes      20%         notify-and-disable    --
excessive-multicasts                 yes      25000 pps   notify-and-disable    40
excessive-link-flaps                 yes      7           notify                --
excessive-oversize-packets           yes      30          notify-and-disable    --
excessive-fragments                 yes      30          notify-and-disable    --
excessive-crc-errors                 yes      30          notify-and-disable    --
```

excessive-late-collisions	yes	30	notify-and-disable	--
excessive-collisions	yes	30	notify-and-disable	--
excessive-tx-drops	yes	30	notify-and-disable	--
excessive-alignment-errors	yes	30	notify-and-disable	--

Showing information for a particular fault monitoring profile:

```
switch# show fault-monitor profile noisy-ports
-----
Fault monitor profile: noisy-ports
-----
Auto
Fault                               Enabled  Threshold  Action                Enable
-----
excessive-broadcasts                yes      5%          notify-and-disable    --
excessive-multicasts                yes      1000 pps    notify-and-disable    --
excessive-link-flaps                yes      7           notify-and-disable    --
excessive-oversize-packets          yes      25          notify-and-disable    --
excessive-fragments                 yes      25          notify-and-disable    --
excessive-crc-errors                yes      25          notify-and-disable    --
excessive-late-collisions            yes      25          notify-and-disable    --
excessive-collisions                yes      25          notify-and-disable    --
excessive-tx-drops                  yes      25          notify-and-disable    --
excessive-alignment-errors          yes      25          notify-and-disable    --
```

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	Manager (#)	Administrators or local user group members with execution rights for this command.

## show interface fault-monitor profile

show interface [<INTERFACE>|<IF-RANGE>] fault-monitor profile

### Description

Shows fault monitoring profile configuration information for all or specific interfaces.

Parameter	Description
<INTERFACE>	Specifies a single interface.
<IF-RANGE>	Specifies a interface range,

### Example

Showing all interfaces with applied fault monitoring profiles:

```
switch# show interface fault-monitor profile
```

```
-----  
Port      Fault Monitor Profile  
-----
```

```
1/1/1     noisy-ports  
1/1/2     quiet-ports  
1/1/4     quiet-ports  
1/1/5     noisy-ports  
1/1/6     noisy-ports  
1/1/7     quiet-ports
```

Showing a range of interfaces with applied fault monitoring profiles:

```
switch# show interface 1/1/1-1/1/2,1/1/6 fault-monitor profile
```

```
-----  
Port      Fault Monitor Profile  
-----
```

```
1/1/1     noisy-ports  
1/1/2     quiet-ports  
1/1/6     noisy-ports
```

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	Manager (#)	Administrators or local user group members with execution rights for this command.

## show interface fault-monitor status

```
show interface [<INTERFACE>|<IF-RANGE>] fault-monitor status
```

### Description

Shows active fault information for all or specific interfaces.

Parameter	Description
<INTERFACE>	Specifies a single interface.
<IF-RANGE>	Specifies a interface range,

### Example

Showing active fault information for all interfaces with applied fault monitoring profiles:

```
switch# show interface fault-monitor status
```

Port	Fault	Fault Elapsed Time	Port State	Time Left
1/1/1	excessive-broadcasts	Tue Apr 14 14:29:09 UTC 2020	down	60
1/1/2	excessive-oversize-packets	Tue Apr 16 14:29:09 UTC 2020	down	--

Showing active fault information for a range of interfaces with applied fault monitoring profiles:

```
switch# show interface 1/3/1,1/3/3 fault-monitor status
```

Port	Fault	Occurring Since	Port State	Time Left
1/1/4	excessive-broadcasts	Tue Apr 14 14:29:09 UTC 2020	down	60

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	Manager (#)	Administrators or local user group members with execution rights for this command.

## show running-config

```
show running-config [interface <IFNAME> | current-context | all]
```

### Description

Displays the fault-monitor profile configurations and profile-name applied to an interface.

Parameter	Description
interface <IFNAME>	Specifies a single interface.
current-context	Displays only current context information.
all	Displays all options in the running config.

### Example

Showing the running configuration for the fault monitoring profiles:

```
switch# show running-config
fault-monitor profile noisy-ports
  excessive-broadcasts
  excessive-broadcasts threshold pps 10000
  excessive-broadcasts action notify-and-disable auto-enable 2000
```

```

excessive-multicasts
excessive-multicasts threshold pps 10000
excessive-link-flaps
excessive-link-flaps action notify-and-disable auto-enable 2000
interface 1/1/1
  apply fault-monitor profile noisy-ports

```

Showing the running configuration with the all option:

```

switch# show running-config all
fault-monitor profile noisy-ports
  excessive-broadcasts
  excessive-broadcasts threshold pps 10000
  excessive-broadcasts action notify-and-disable auto-enable 2000
  excessive-multicasts
  excessive-multicasts threshold pps 10000
  excessive-multicasts action notify
  excessive-link-flaps
  excessive-link-flaps threshold count 7
  excessive-link-flaps action notify-and-disable auto-enable 2000
no excessive-oversize-packets
excessive-oversize-packets threshold value 25
excessive-oversize-packets action notify
no excessive-fragments
excessive-fragments threshold value 25
excessive-fragments action notify
no excessive-crc-errors
excessive-crc-errors threshold value 25
excessive-crc-errors action notify
no excessive-late-collisions
excessive-late-collisions threshold value 25
excessive-late-collisions action notify
no excessive-collisions
excessive-collisions threshold value 25
excessive-collisions action notify
no excessive-tx-drops
excessive-tx-drops threshold value 25
excessive-tx-drops action notify
no excessive-alignment-errors
excessive-alignment-errors threshold value 25
excessive-alignment-errors action notify

```

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	Manager (#)	Administrators or local user group members with execution rights for this command.

## threshold

```

<FAULT> threshold value <VALUE>
no <FAULT> threshold

excessive-link-flaps threshold count <COUNT>
no excessive-link-flaps threshold

{excessive-broadcasts | excessive-multicasts}
  threshold {percent <BW-PERCENT> | pps <PPS>}
no {excessive-broadcasts | excessive-multicasts} threshold

```

## Description

Within the selected fault monitor profile context, configures the fault threshold value for the profile. The `no` form of this command sets the threshold to its default value.

Parameter	Description
<FAULT>	Available fault names: excessive-crc-errors excessive-oversize-packets excessive-fragments excessive-tx-drops excessive-collisions excessive-late-collisions excessive-alignment-errors excessive-alignment-errors
threshold value <VALUE>	Specifies the fault threshold value. Default: 25.
threshold count <COUNT>	Specifies the fault threshold count. Default threshold count: 7.
threshold percent <BW-PERCENT>	Specifies the fault threshold bandwidth percentage. Range: 1 to 100. Default: 5.
threshold pps <PPS>	Specifies the fault threshold PPS (packets per second). Range: 1 to 195312500.

If excessive-broadcast or excessive-multicast faults are configured with the threshold higher than the `rate-limit` threshold, the following occurs:

- Fault reporting still happens as the port has actually received packets at a rate that violated its threshold.
- Traffic gets shaped as per `rate-limit` configuration and any packet exceeding the `rate-limit` threshold gets dropped.

## Examples

Configuring with threshold values:

```

switch(config-fault-monitor-profile)# excessive-oversize-packets threshold value 40
switch(config-fault-monitor-profile)# excessive-crc-errors threshold value 35
switch(config-fault-monitor-profile)# excessive-fragments threshold value 50
switch(config-fault-monitor-profile)# excessive-tx-drops threshold value 20
switch(config-fault-monitor-profile)# excessive-collisions threshold value 40
switch(config-fault-monitor-profile)# excessive-late-collisions threshold value 30

```



```
switch(config-fault-monitor-profile)# excessive-alignment-errors threshold value 50
```

Configuring with a threshold count:

```
switch(config-fault-monitor-profile)# excessive-link-flaps threshold count 10
```

Configuring with threshold percents and PPS:

```
switch(config-fault-monitor-profile)# excessive-broadcasts threshold percent 40
switch(config-fault-monitor-profile)# excessive-multicasts threshold pps 10000
```

Removing the configured threshold value for the faults:

```
switch(config-fault-monitor-profile)# no excessive-oversize-packets threshold
switch(config-fault-monitor-profile)# no excessive-crc-errors threshold
switch(config-fault-monitor-profile)# no excessive-fragments threshold
switch(config-fault-monitor-profile)# no excessive-tx-drops threshold
switch(config-fault-monitor-profile)# no excessive-collisions threshold
switch(config-fault-monitor-profile)# no excessive-late-collisions threshold
switch(config-fault-monitor-profile)# no excessive-alignment-errors threshold
switch(config-fault-monitor-profile)# no excessive-link-flaps threshold
switch(config-fault-monitor-profile)# no excessive-broadcasts threshold
switch(config-fault-monitor-profile)# no excessive-multicasts threshold
```

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	config-fault-monitor-profile	Administrators or local user group members with execution rights for this command.

## vsx-sync (fault monitor)

```
vsx-sync
no vsx-sync
```

### Description

Within the selected fault monitor profile context, configures VSX synchronization for the selected fault monitoring profile.

The `no` form of this command removes the VSX synchronization for a fault monitoring profile.

### Example

Configuring VSX synchronization for a fault monitoring profile:

```
switch(config-fault-monitor-profile) # vsx-sync
```

## Command History

Release	Modification
10.07.0030	Command introduced

## Command Information

Platforms	Command context	Authority
4100i 8400	config-fault-monitor-profile	Administrators or local user group members with execution rights for this command.

The `auditors` group enables administrators to create users that can perform auditing tasks without allowing those users the authority to view or change the switch configuration.

As is the case for other users, auditors can access the switch using the Web UI, REST API, or the CLI.

## Auditing tasks (CLI)

When you log on to the switch CLI as a user with auditor rights, you have access to the auditor command context only.

```
auditor>
```

The tasks that can be performed by auditors are as follows. The commands listed are the only commands auditors can execute other than session commands like `print`, `list`, and `exit`. However, auditors can use all command options except as noted. See the command description for each command for complete information about the command.

Task	Command name	Example
Show event log contents	<code>show events</code>	<code>show events -a -r</code>
Show local accounting log contents	<code>show accounting log</code>	<code>show accounting log last 10</code>
Copy command output to a remote server or to a local USB drive.	<code>copy command-output</code>	<code>copy command-output "show events -a -r" tftp://10.100.0.12/file</code>

When using the `copy command-output` command, users with auditor rights can specify the following commands only:

```
show accounting log  
show events
```

## Auditing tasks (Web UI)

Auditors have access to the Log page only. When you log on to the switch Web UI as a user with auditor rights, the Log page is displayed.

From the Log page, you can view and export event log entries.

The Web UI does not provide access to the accounting logs.

## REST requests and accounting logs

All REST requests—including GET requests—are logged to the accounting (audit) log.

The URI of the REST API resource for accounting logs is the following:

`/rest/v10.04/logs/audit`

In an accounting log entry for a REST request:

- `service=https-server` indicates that the log entry is a result of a REST API request or a Web UI action.
- The string value of `data` identifies the REST API request that was executed.

For more information about accounting log entries, see the description of the `show accounting log` CLI command.

### Accessing Aruba Support

Aruba Support Services	<a href="https://www.arubanetworks.com/support-services/">https://www.arubanetworks.com/support-services/</a>
Aruba Support Portal	<a href="https://asp.arubanetworks.com/">https://asp.arubanetworks.com/</a>
North America telephone	1-800-943-4526 (US & Canada Toll-Free Number) +1-408-754-1200 (Primary - Toll Number) +1-650-385-6582 (Backup - Toll Number - Use only when all other numbers are not working)
International telephone	<a href="https://www.arubanetworks.com/support-services/contact-support/">https://www.arubanetworks.com/support-services/contact-support/</a>

Be sure to collect the following information before contacting Support:

- Technical support registration number (if applicable)
- Product name, model or version, and serial number
- Operating system name and version
- Firmware version
- Error messages
- Product-specific reports and logs
- Add-on products or components
- Third-party products or components

#### Other useful sites

Other websites that can be used to find information:

Airheads social forums and Knowledge Base	<a href="https://community.arubanetworks.com/">https://community.arubanetworks.com/</a>
Software licensing	<a href="https://lms.arubanetworks.com/">https://lms.arubanetworks.com/</a>
End-of-Life information	<a href="https://www.arubanetworks.com/support-services/end-of-life/">https://www.arubanetworks.com/support-services/end-of-life/</a>
Aruba software and documentation	<a href="https://asp.arubanetworks.com/downloads">https://asp.arubanetworks.com/downloads</a>

### Accessing Updates

You can access updates from the Aruba Support Portal or the HPE My Networking Website.

#### Aruba Support Portal

<https://asp.arubanetworks.com/downloads>

If you are unable to find your product in the Aruba Support Portal, you may need to search My Networking, where older networking products can be found:

## My Networking

<https://www.hpe.com/networking/support>

To view and update your entitlements, and to link your contracts and warranties with your profile, go to the Hewlett Packard Enterprise Support Center **More Information on Access to Support Materials** page:

<https://support.hpe.com/portal/site/hpsc/aae/home/>

Access to some updates might require product entitlement when accessed through the Hewlett Packard Enterprise Support Center. You must have an HP Passport set up with relevant entitlements.

Some software products provide a mechanism for accessing software updates through the product interface. Review your product documentation to identify the recommended software update method.

To subscribe to eNewsletters and alerts:

<https://asp.arubanetworks.com/notifications/subscriptions> (requires an active Aruba Support Portal (ASP) account to manage subscriptions). Security notices are viewable without an ASP account.

## Warranty Information

To view warranty information for your product, go to <https://www.arubanetworks.com/support-services/product-warranties/>.

## Regulatory Information

To view the regulatory information for your product, view the *Safety and Compliance Information for Server, Storage, Power, Networking, and Rack Products*, available at <https://www.hpe.com/support/Safety-Compliance-EnterpriseProducts>

### Additional regulatory information

Aruba is committed to providing our customers with information about the chemical substances in our products as needed to comply with legal requirements, environmental data (company programs, product recycling, energy efficiency), and safety information and compliance data, (RoHS and WEEE). For more information, see <https://www.arubanetworks.com/company/about-us/environmental-citizenship/>.

## Documentation Feedback

Aruba is committed to providing documentation that meets your needs. To help us improve the documentation, send any errors, suggestions, or comments to Documentation Feedback ([docsfeedback-switching@hpe.com](mailto:docsfeedback-switching@hpe.com)). When submitting your feedback, include the document title, part number, edition, and publication date located on the front cover of the document. For online help content, include the product name, product version, help edition, and publication date located on the legal notices page.