AOS-CX 10.07 Fundamentals Guide

6300, 6400 Switch Series

a Hewlett Packard Enterprise company

Part Number: 5200-7851
Published: April 2021
Edition: 1

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.

Acknowledgments

Bluetooth is a trademark owned by its proprietor and used by Hewlett Packard Enterprise under license.

Chapter 1
About this document

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 6300 Switch Series (JL658A, JL659A, JL660A, JL661A, JL662A, JL663A, JL664A, JL665A, JL666A, JL667A, JL668A, JL762A)
- Aruba 6400 Switch Series (JL741A, R0X26A, R0X27A, R0X29A, R0X30A)

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 (`[ ]`).
example-text	In code and screen examples, indicates text entered by a user.
Any of the following: ■ *<example-text>* ■ <example-text> ■ *example-text* ■ *example-text*	Identifies a placeholder—such as a parameter or a variable—that you must substitute with an actual value in a command or in code: ■ For output formats where italic text cannot be displayed, variables are enclosed in angle brackets (< >). Substitute the text—including the enclosing angle brackets—with an actual value. ■ 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.
|	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.
{ }	Braces. Indicates that at least one of the enclosed items is required.

Convention	Usage
`[ ]`	Brackets. Indicates that the enclosed item or items are optional.
... or . . .	Ellipsis: ■ In code and screen examples, a vertical or horizontal ellipsis indicates an omission of information. ■ 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.

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 6300 Switch Series

- *member*: Member number of the switch in a Virtual Switching Framework (VSF) stack. Range: 1 to 10. The primary switch is always member 1. If the switch is not a member of a VSF stack, then member is 1.

- *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 member 1.

On the 6400 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 1/1 and 1/2.
  - Line modules are on the front of the switch starting in slot 1/3.
- *port*: Physical number of a port on a line module.

For example, the logical interface `1/3/4` in software is associated with physical port 4 in slot 3 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.

AOS-CX is a new, modern, fully programmable operating system built using a database-centric design that ensures higher availability and dynamic software process changes for reduced downtime. In addition to robust hardware reliability, the AOS-CX operating system includes additional software elements not available with traditional systems, including:

- Automated visibility to help IT organizations scale: The Aruba Network Analytics Engine allows IT to monitor and troubleshoot network, system, application, and security-related issues easily through simple scripts. This engine comes with a built-in time series database that enables customers and developers to create software modules that allow historical troubleshooting, as well as analysis of historical trends to predict and avoid future problems due to scale, security, and performance bottlenecks.
- Programmability simplified: A switch that is running the AOS-CX operating system is fully programmable with a built-in Python interpreter as well as REST-based APIs, allowing easy integration with other devices both on premise and in the cloud. This programmability accelerates IT organization understanding of and response to network issues. The database holds all aspects of the configuration, statistics, and status information in a highly structured and fully defined form.
- Faster resolution with network insights: With legacy switches, IT organizations must troubleshoot problems after the fact, using traditional tools like CLI and SNMP, augmented by separate, expensive monitoring, analytics, and troubleshooting solutions. These capabilities are built in to the AOS-CX operating system and are extensible.
- High availability: For switches that support active and standby management modules, the AOS-CX database can synchronize data between active and standby modules and maintain current configuration and state information during a failover to the standby management module.
- Ease of roll-back to previous configurations: The built-in database acts as a network record, enabling support for multiple configuration checkpoints and the ability to roll back to a previous configuration checkpoint.

AOS-CX system databases

The AOS-CX operating system is a modular, database-centric operating system. Every aspect of the switch configuration and state information is modeled in the AOS-CX switch configuration and state database, including the following:

- Configuration information
- Status of all features
- Statistics

The AOS-CX operating system also includes a time series database, which acts as a built-in network record. The time series database makes the data seamlessly available to Aruba Network Analytics Engine agents that use rules that evaluate network conditions over time. Time-series data about the resources monitored by agents are automatically collected and presented in graphs in the switch Web UI.

Aruba Network Analytics Engine introduction

The Aruba Network Analytics Engine is a first-of-its-kind built-in framework for network assurance and remediation. Combining the full automation and deep visibility capabilities of the AOS-CX operating system, this unique framework enables monitoring, collecting network data, evaluating conditions, and taking corrective actions through simple scripting agents.

This engine is integrated with the AOS-CX system configuration and time series databases, enabling you to examine historical trends and predict future problems due to scale, security, and performance bottlenecks. With that information, you can create software modules that automatically detect such issues and take appropriate actions.

With the faster network insights and automation provided by the Aruba Network Analytics Engine, you can reduce the time spent on manual tasks and address current and future demands driven by Mobility and IoT.

AOS-CX CLI

The AOS-CX CLI is an industry standard text-based command-line interface with hierarchical structure designed to reduce training time and increase productivity in multivendor installations.

The CLI gives you access to the full set of commands for the switch while providing the same password protection that is used in the Web UI. You can use the CLI to configure, manage, and monitor devices running the AOS-CX operating system.

Aruba CX mobile app

The Aruba CX mobile app enables you to use a mobile device to configure or access a supported ArubaOS-CX switch. You can connect to the switch through Bluetooth or Wi-Fi.

You can use this application to do the following:

- Connect to the switch for the first time and configure basic operational settings—all without requiring you to connect a terminal emulator to the console port.
- View and change the configuration of individual switch features or settings.
- Manage the running configuration and startup configuration of the switch, including the following:
  - Transferring files between the switch and your mobile device
  - Sharing configuration files from your mobile device
  - Copying the running configuration to the startup configuration
- Access the switch CLI.

For more information about the Aruba CX mobile app, see:

www.arubanetworks.com/products/networking/switches/cx-mobileapp.

Aruba NetEdit

Aruba NetEdit enables the automation of multidevice configuration change workflows without the overhead of programming.

The key capabilities of NetEdit include the following:

- Intelligent configuration with validation for consistency and compliance
- Time savings by simultaneously viewing and editing multiple configurations
- Customized validation tests for corporate compliance and network design
- Automated large-scale configuration deployment without programming

- Ability to track changes to hardware, software, and configurations (whether made through NetEdit or directly on the switch) with automated versioning

For more information about Aruba NetEdit, search for NetEdit at the following website:

www.hpe.com/support/hpesc

Ansible modules

Ansible is an open-source IT automation platform.

Aruba publishes a set of Ansible configuration management modules designed for switches running AOS-CX software. The modules are available from the following places:

- The `arubanetworks.aoscx_role` role in the Ansible Galaxy at: https://galaxy.ansible.com/arubanetworks/aoscx_role
- The `aoscx-ansible-role` at the following GitHub repository: https://github.com/aruba/aoscx-ansible-role

AOS-CX Web UI

The Web UI gives you quick and easy visibility into what is happening on your switch, providing faster problem detection, diagnosis, and resolution. The Web UI provides dashboards and views to monitor the status of the switch, including easy to read indicators for: power supply, temperature, fans, CPU use, memory use, log entries, system information, firmware, interfaces, VLANs, and LAGs. In addition, you use the Web UI to access the Network Analytics Engine, run certain diagnostics, and modify some aspects of the switch configuration.

AOS-CX REST API

Switches running the AOS-CX software are fully programmable with a REST (REpresentational State Transfer) API, allowing easy integration with other devices both on premises and in the cloud. This programmability—combined with the Aruba Network Analytics Engine—accelerates network administrator understanding of and response to network issues.

The AOS-CX REST API enables programmatic access to the AOS-CX configuration and state database at the heart of the switch. By using a structured model, changes to the content and formatting of the CLI output do not affect the programs you write. And because the configuration is stored in a structured database instead of a text file, rolling back changes is easier than ever, thus dramatically reducing a risk of downtime and performance issues.

The AOS-CX REST API is a web service that performs operations on switch resources using HTTPS `POST`, `GET`, `PUT`, and `DELETE` methods.

A switch resource is indicated by its Uniform Resource Identifier (URI). A URI can be made up of several components, including the host name or IP address, port number, the path, and an optional query string. The AOS-CX operating system includes the AOS-CX REST API Reference, which is a web interface based on the Swagger UI. The AOS-CX REST API Reference provides the reference documentation for the REST API, including resources URIs, models, methods, and errors. The AOS-CX REST API Reference shows most of the supported read and write methods for all switch resources.

In-band and out-of-band management

Management communications with a managed switch can be either of the following:

In band

In-band management communications occur through ports on the line modules of the switch, using common communications protocols such as SSH and SNMP.

When you use an in-band management connection, management traffic from that connection uses the same network infrastructure as user data. User data uses the `data plane`, which is responsible for moving data from source to destination. Management traffic that uses the data plane is more likely to be affected by traffic congestion and other issues affecting the user network.

Out of band

OOBM (out-of-band management) communications occur through a dedicated serial or USB console port or though a dedicated networked management port.

OOBM operates on a `management plane` that is separate from the `data plane` used by data traffic on the switch and by in-band management traffic. That separation means that OOBM can continue to function even during periods of traffic congestion, equipment malfunction, or attacks on the network. In addition, it can provide improved switch security: a properly configured switch can limit management access to the management port only, preventing malicious attempts to gain access through the data ports.

Networked OOBM typically occurs on a management network that connects multiple switches. It has the added advantage that it can be done from a central location and does not require an individual physical cable from the management station to the console port of each switch.

SNMP-based management support

The AOS-CX operating system provides SNMP read access to the switch. SNMP support includes support of industry-standard MIB (Management Information Base) plus private extensions, including SNMP events, alarms, history, statistics groups, and a private alarm extension group. SNMP access is disabled by default.

User accounts

To view or change configuration settings on the switch, users must log in with a valid account. Authentication of user accounts can be performed locally on the switch, or by using the services of an external TACACS+ or RADIUS server.

Two types of user accounts are supported:

- Operators: Operators can view configuration settings, but cannot change them. No operator accounts are created by default.
- Administrators: Administrators can view and change configuration settings. A default locally stored administrator account is created with username set to **admin** and no password. You set the administrator account password as part of the initial configuration procedure for the switch.

Perform the initial configuration of a factory default switch using one of the following methods:

- Load a switch configuration using zero-touch provisioning (ZTP). When ZTP is used, the configuration is loaded from a server automatically when the switch booted from the factory default configuration.
- Connect to the switch wirelessly with a mobile device through Bluetooth, and use the Aruba CX Mobile App to deploy an initial configuration from a provided template. The template you choose during the deployment process determines how the management interface is configured. Optionally, as the final deployment step, you can select to import the switch into NetEdit through a WiFI connection to the NetEdit server.

Alternatively, you can use the Aruba CX Mobile App to manually configure switch settings and features for a subset of the features you can configure using the CLI. You can also access the CLI through the mobile application.

- Connect the management port on the switch to your network, and then use SSH client software to reach the switch from a computer connected to the same network. This requires that a DHCP server is installed on the network. Configure switch settings and features by executing CLI commands.
- Connect a computer running terminal emulation software to the console port on the switch. Configure switch settings and features by executing CLI commands.

Initial configuration using ZTP

Zero Touch Provisioning (ZTP) configures a switch automatically from a remote server.

Prerequisites

- The switch must be in the factory default configuration.

Do not change the configuration of the switch from its factory default configuration in any way, including by setting the administrator password.

- Your network administrator or installation site coordinator must provide a Category 6 (Cat6) cable connected to the network that provides access to the servers used for Zero Touch Provisioning (ZTP) operations.

Procedure

1. Connect the network cable to the out-of-band management port on the switch. If your network administrator or installation site coordinator has instructed you to connect network cable to a data port, connect the cable to that data port instead.

   See the *Installation Guide* for switch to determine the location of the switch ports.
2. If the switch is powered on, power off the switch.
3. Power on the switch. During the ZTP operation, the switch might reboot if a new firmware image is being installed. ZTP goes to "Failed" state if the switch receives DHCP IP for vlan1 and does not receive any ZTP options within 60 seconds.

Initial configuration using the Aruba CX mobile app

This procedure describes how to use your mobile device to connect to the Bluetooth interface of the switch to connect to the switch for the first time so that you can configure basic operational settings using the Aruba CX mobile app.

Prerequisites

- You have obtained the USB Bluetooth adapter that was shipped with the switch. Information about the make and model of the supported adapter is included in the information about the Aruba CX mobile app in the Apple Store or Google Play.
- The Aruba CX mobile app must be installed on your mobile device.
- Bluetooth must be enabled on your mobile device.
- Your mobile device must be within the communication range of the Bluetooth adapter.
- If you are planning to import the switch into NetEdit, your mobile device must be able to use a Wi-Fi connection—not Bluetooth—to access the NetEdit server.

If your mobile device does not support simultaneous Bluetooth and Wi-Fi connections, you must use the NetEdit interface to import the switch at a later time. You can use the **Devices** tab to display the IP address of the switches you configured using your mobile device.

- The switch must be installed and powered on, with the network operating system boot sequence complete.

For information about installing and powering on the switch, see the *Installation Guide* for the switch.

Because you are using this mobile application to configure the switch through the Bluetooth interface, it is not necessary to connect a console to the switch.

- Bluetooth and USB must be enabled on the switch. On switches shipped from the factory, Bluetooth and USB are enabled by default.

Procedure

1. Install the USB Bluetooth adapter in the USB port of the switch.

   For switches that have multiple management modules, you must install the USB Bluetooth adapter in the USB port of the active management module. Typically, the active management module is the module in slot 5. On the 6400, the active management module is typically installed in slot 1. When configuring a stack, a USB Bluetooth adapter must be installed on each 6300 switch in the stack.

   Switches shipped from the factory have both USB and Bluetooth enabled by default.

   For information about the location of the USB port on the switch, see the *Installation Guide* for the switch.
2. Use the Bluetooth settings on your mobile device to pair and connect the switch to your mobile device.

   If you are in range of multiple Bluetooth devices, more than one device is displayed on the list of available devices. Switches running the AOS-CX operating system are displayed in the following format:

   `Switch_model - Serial_number`

   For example: `8325-987654X1234567` or `8320-AB12CDE123`

   A switch supports one active Bluetooth connection at a time.

   On some Android devices, you might need to change the settings of the paired device to specify that it be used for Internet access.

3. Open the Aruba CX mobile app on your mobile device.


;" />

   The application attempts to connect to the switch using the switch Bluetooth IP address and the default switch login credentials. The **Home** screen of the application shows the status of the connection to the switch:

   - If the login attempt was successful, the Bluetooth icon is displayed and the status message shows the Bluetooth IP address of the switch. In addition, the connection graphic is green. You can continue to the next step.
   - If the login attempt was not successful, but a response was received, the Bluetooth icon is displayed, but the status message is: `Login Required`. You can continue to the next step. When you tap one of the tiles, you will be prompted for login credentials.
   - If the login attempt did not receive a response, the Bluetooth icon is not displayed, and the status message is: `No Connection`.

4. Create the initial switch configuration:
   - You can deploy an initial configuration to the switch. Through this process, you supply the information required by a configuration template that you choose from a list of templates provided by the application. Then you deploy the configuration to the switch and, optionally, import the switch into NetEdit.

When you deploy a switch configuration, it becomes the running configuration, replacing the entire existing configuration of the switch. All changes previously made to the factory default configuration are overwritten.
If you plan to both deploy a switch configuration and customize the configuration of switch features, deploy the initial configuration first.

   To deploy an initial switch configuration, tap: **Initial Config** and follow the instructions in the application.

   - Alternatively, you can complete the initial configuration of the switch by tapping **Modify Config** and then selecting the features and settings to configure.
   - You can also use the **Modify Config** feature to configure some switch features after the initial configuration is complete. For more information about what you can configure using the Aruba CX mobile app, see the online help for the application.

Troubleshooting Bluetooth connections

Bluetooth connection IP addresses

The Bluetooth connection uses IP addresses in the 192.168.99.0/24 subnet.

Switch

192.168.99.1

Mobile device

192.168.99.10

Bluetooth is connected but the switch is not reachable

Symptom

The mobile device settings indicate that the device is connected to the switch through Bluetooth. However, the mobile application indicates that the switch is not reachable.

Solution 1

Cause

The mobile device is paired with a different nearby switch.

Action

1. Verify the model number and serial number of the switch to which you are attempting to connect.
2. Use the Bluetooth settings on your mobile device to pair and connect the switch to your mobile device.

   If you are in range of multiple Bluetooth devices, more than one device is displayed on the list of available devices. Switches running the AOS-CX operating system are displayed in the following format:
   `Switch_model-Serial_number`

   For example: `8325-987654X1234567` or `8320-AB12CDE123`

   A switch supports one active Bluetooth connection at a time.

   On some Android devices, you might need to change the settings of the paired device to specify that it be used for Internet access.

Solution 2

Cause

The mobile device is connected to a different network—such as through a Wi-Fi connection—that conflicts with the subnet used for the switch Bluetooth connection.

Action

Disconnect the mobile device from the network that is using the conflicting subnet.

For example, use the mobile device settings to turn off or disable Wi-Fi. If you choose to disable Wi-Fi on the mobile device, and you are not able to access cellular service, you will not be able to connect to the NetEdit server to import the switch, but you can still deploy a switch configuration.

Bluetooth is not connected

Symptom

Your mobile device cannot establish a Bluetooth connection to the switch.

Solution 1

Cause

Bluetooth is not enabled on your mobile device.

Action

- Use your mobile device settings application to enable Bluetooth.
- Use the Bluetooth settings on your mobile device to pair and connect the switch to your mobile device.

If you are in range of multiple Bluetooth devices, more than one device is displayed on the list of available devices. Switches running the AOS-CX operating system are displayed in the following format:
`Switch_model-Serial_number`

For example: `8325-987654X1234567` or `8320-AB12CDE123`

A switch supports one active Bluetooth connection at a time.

On some Android devices, you might need to change the settings of the paired device to specify that it be used for Internet access.

Solution 2

Cause

Your mobile device is not within the broadcast range of the Bluetooth adapter.

Action

Move closer to the switch.

Devices can communicate through Bluetooth when they are close, typically within a few feet of each other.

Solution 3

Cause

Your mobile device is not paired with the switch.

Action

1. Use your mobile device settings application to enable Bluetooth.
2. Use the Bluetooth settings on your mobile device to pair and connect the switch to your mobile device.

   If you are in range of multiple Bluetooth devices, more than one device is displayed on the list of available devices. Switches running the AOS-CX operating system are displayed in the following format:
   `Switch_model-Serial_number`

   For example: `8325-987654X1234567` or `8320-AB12CDE123`

   A switch supports one active Bluetooth connection at a time.
3. On some Android devices, you might need to change the settings of the paired device to specify that it be used for Internet access.

Solution 4

Cause

Bluetooth is not enabled on the switch.

New switches are shipped from the factory with the USB port and Bluetooth enabled. However, an installed switch might have been configured to disable Bluetooth or disable the USB port, which the USB Bluetooth adapter uses.

Action

Use a different CLI connection to enable Bluetooth on the switch.

- Use the `show bluetooth` CLI command to show the Bluetooth configuration and the status of the Bluetooth adapter.
- To enable the USB port, enter the CLI command: `usb`
- An inserted USB drive must be mounted each time the switch boots or fails over to a different management module. To mount the drive, enter the CLI command: `usb mount`
- To enable Bluetooth, enter the CLI command: `bluetooth enable`

Solution 5

Cause

Another mobile device has already connected to the switch through Bluetooth. This cause is likely if your device is repeatedly disconnected within 1-2 seconds of establishing a connection.

Action

1. Use a different CLI connection to see if there is another device connected:

   Use the `show bluetooth` CLI command to show the Bluetooth configuration and the status of the Bluetooth adapter.
2. Either disconnect the other device or use that device to communicate with the switch.

   A switch can use Bluetooth to connect to one mobile device at a time.

Solution 6

Cause

The switch has been restarted since the mobile device was last paired with the switch, and the device is having difficulty establishing the Bluetooth connection.

Action

1. Use the Bluetooth mobile device settings to forget the switch device.
2. Use your mobile device settings application to disable Bluetooth.
   Use your mobile device settings application to enable Bluetooth.

   If you are in range of multiple Bluetooth devices, more than one device is displayed on the list of available devices. Switches running the AOS-CX operating system are displayed in the following format:
   `Switch_model-Serial_number`

   For example: `8325-987654X1234567` or `8320-AB12CDE123`

   A switch supports one active Bluetooth connection at a time.

   On some Android devices, you might need to change the settings of the paired device to specify that it be used for Internet access.

Solution 7

Cause

The USB Bluetooth adapter is not installed in the switch.

If the switch has multiple management modules, the USB Bluetooth adapter might be installed in the management module that is not the active management module.

Action

Install the USB Bluetooth adapter in the USB port of the switch.

For switches that have multiple management modules, you must install the USB Bluetooth adapter in the USB port of the active management module. Typically, for new switches, the active management module is the module in slot 5 (Aruba 8400 switches) or slot 1 (Aruba 6400 switches).

For information about the location of the USB port on the switch, see the *Installation Guide* for the switch.

Solution 8

Cause

A problem occurred with the Bluetooth feature on the switch. For example, the software daemon was stopped and then restarted.

Action

1. Use a different connection to the switch CLI to disable and then enable Bluetooth.

```
switch(config)# bluetooth disable
switch(config)# bluetooth enable
```

2. Use the Bluetooth mobile device settings to forget the switch device.
3. Use your mobile device settings application to disable Bluetooth.
4. Use your mobile device settings application to enable Bluetooth.
5. Use your mobile device settings application to enable Bluetooth.

   If you are in range of multiple Bluetooth devices, more than one device is displayed on the list of available devices. Switches running the AOS-CX operating system are displayed in the following format:
   Switch_model-Serial_number

   For example: `8325-987654X1234567` or `8320-AB12CDE123`

   A switch supports one active Bluetooth connection at a time.

   On some Android devices, you might need to change the settings of the paired device to specify that it be used for Internet access.

Solution 9

Cause

A switch that is member of a stack (but is not the master switch), has a USB Bluetooth adapter installed, but mobile application has lost contact with that switch.

Action

Remove and then reinstall the USB Bluetooth adapter.

Do not remove the USB Bluetooth adapter from the master switch.

Initial configuration using the CLI

This procedure describes how to connect to the switch for the first time and configure basic operational settings using the CLI. In this procedure, you use a computer to connect to the switch using the either the console port or management port.

Procedure

1. Connect to the console port or the management port.
2. Log into the switch for the first time.
3. Configure switch time using the NTP client.

Connecting to the console port

Prerequisites

- A switch installed as described in its hardware installation guide.
- A computer with terminal emulation software.
- A JL448A Aruba X2 C2 RJ45 to DB9 console cable. (6400 only), or a USB-C cable (6300/6400).

Procedure

1. Connect the console port on the switch to the serial port on the computer using a console cable, or connect the USB-C port on the switch to the USB-C port on the computer using a USB-C cable.
2. Start the terminal emulation software on the computer and configure a new serial session with the following settings:
   - Speed: 115200 bps
   - Data bits: 8
   - Stop bits: 1
   - Parity: None
   - Flow control: None
3. Start the terminal emulation session.
4. Press **Enter** once. If the connection is successful, you are prompted to login.

Connecting to the management port

Prerequisites

- Two Ethernet cables
- SSH client software

Procedure

1. By default, the management interface is set to automatically obtain an IP address from a DHCP server, and SSH support is enabled. If there is no DHCP server on your network, you must configure a static address on the management interface:
   a. Connect to the console port
   b. Configure the management interface.
2. Use an Ethernet cable to connect the management port to your network.
3. Use an Ethernet cable to connect your computer to the same network.
4. Start your SSH client software and configure a new session using the address assigned to the management interface. (If the management interface is set to operate as a DHCP client, retrieve the IP address assigned to the management interface from your DHCP server.)
5. Start the session. If the connection is successful, you are prompted to log in.

Configure using DHCP or static IP

Users can use any data ports for in-band management purposes. IP DHCP is supported on interface VLAN 1 only. All switch ports are part of access VLAN 1 by default. Static IP address and IP DHCP configuration can co-exist on VLAN 1, however static addresses take precedence whenever configured.

DHCP Configuration

```
switch#: config
switch(config)#: vlan 1
Switch(config-vlan-1)#: description Management VLAN
Switch(config-vlan-1)#: end
Switch#
!
Switch(config)#: interface 1/1/1
Switch(config-if)#: description IN-BAND Management Port
```

```
Switch(config-if)#: vlan access 1
Switch(config-if)#: no shutdown
Switch(config-if)#: end
Switch#
!
Switch(config)#: interface vlan 1
Switch(config-if-vlan)#: description IN-BAND Management Interface
Switch(config-if-vlan)#: ip dhcp
Switch(config-if-vlan)#: no shutdown
Switch(config-if-vlan)#: end
Switch#
!
```

Without DHCP Configuration

```
switch#: config
switch(config)#: vlan 1
Switch(config-vlan-1)#: description Management VLAN
Switch(config-vlan-1)#: end
Switch#
!
Switch(config)#: interface 1/1/1
Switch(config-if)#: description IN-BAND Management Port
Switch(config-if)#: vlan access 1
Switch(config-if)#: no shutdown
Switch(config-if)#: end
Switch#
!
Switch(config)#: interface vlan 1
Switch(config-if-vlan)#: description IN-BAND Management Interface
Switch(config-if-vlan)#: no ip dhcp
Switch(config-if-vlan)#: ip address 192.168.10.1/24
Switch(config-if-vlan)#: no shutdown
Switch(config-if-vlan)#: end
Switch#
```

Logging into the switch for the first time

The first time you log in to the switch you must use the default administrator account. This account has no password, so you will be prompted on login to define one to safeguard the switch.

Procedure

1. When prompted to log in, specify **admin**. When prompted for the password, press **ENTER**. (By default, no password is defined.)

   For example:

   ```
   switch login: admin
   password:
   ```

2. Define a password for the **admin** account. The password can contain up to 32 alphanumeric characters in the range ASCII 32 to 127, which includes special characters such as asterisk (*), ampersand (&), exclamation point (!), dash (-), underscore (_), and question mark (?).

   For example:

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

3. You are placed into the manager command context, which is identified by the prompt: `switch#`, where `switch` is the model number of the switch. Enter the command `config` to change to the global configuration context `config`.

   For example:

```
switch# config
switch(config)#
```

Setting switch time using the NTP client

Prerequisites

- The IP address or domain name of an NTP server.
- If the NTP server uses authentication, obtain the password required to communicate with the NTP server.

Procedure

1. If the NTP server requires authentication, define the authentication key for the NTP client with the command `ntp authentication`.
2. Configure an NTP server with the command `ntp server`.
3. By default, NTP traffic is sent on the default VRF. If you want to send NTP traffic on the management VRF, use the command `ntp vrf`.
4. Review your NTP configuration settings with the commands `show ntp servers` and `show ntp status`.
5. See the current switch time, date, and time zone with the command `show clock`.

Example

This example creates the following configuration:

- Defines the authentication key **1** with the password **myPassword**.
- Defines the NTP server **my-ntp.mydomain.com** and makes it the preferred server.
- Sets the switch to use the management VRF (**mgmt**) for all NTP traffic.

```
switch(config)# ntp authentication-key 1 md5 myPassword
switch(config)# ntp server my-ntp.mydomain.com key 10 prefer
switch(config)# ntp vrf mgmt
```

Configuring banners

1. Configure the banner that is displayed when a user connects to a management interface. Use the command `banner motd`. For example:

```
switch(config)# banner motd ^
Enter a new banner. Terminate the banner with the delimiter you have chosen.
>> This is an example of a banner text which a connecting user
>> will see before they are prompted for their password.
>>
>> As you can see it may span multiple lines and the input
>> will be terminated when the delimiter character is
>> encountered.^
Banner updated successfully!
```

2. Configure the banner that is displayed after a user is authenticated. Use the command `banner exec`. For example:

```
switch(config)# banner exec &
Enter a new banner. Terminate the banner with the delimiter you have chosen.
>> This is an example of a different banner text. This time
>> the banner entered will be displayed after a user has
>> authenticated.
>>
>> & This text will not be included because it comes after the '&'
Banner updated successfully!
```

Configuring in-band management on a data port

Prerequisites

- A connection to the CLI via either the console port or the management port
- Ethernet cable

Procedure

1. Use an Ethernet cable to connect a data port to your network.
2. Configure a layer 3 interface on the data port.
3. Enable SSH support on the interface (on the default VRF) with the command `ssh server vrf default`.

   For example:
   ```
   switch# config
   switch(config)# ssh server vrf default
   ```
4. Enable the Web UI on the interface (on the default VRF) with the command `https-server vrf default`.

   For example:
   ```
   switch(config)# https-server vrf default
   ```

Using the Web UI

The Web UI is disabled by default. Follow these steps to enable it on the management port and log in. The Web UI is enabled by default on the default VRF.

Prerequisites

- A connection to the switch CLI.

Procedure

1. Log in to the CLI.
2. Switch to `config` context and enable the Web UI on the management port VRF with the command `https-server vrf mgmt`.

   For example:

   ```
   switch# config
   switch(config)# https-server vrf mgmt
   ```

3. Start your web browser and enter the IP address of the management port in the address bar,

   For example: **https://192.168.1.1**
4. The Web UI starts and you are prompted to log in.

Configuring the management interface

Prerequisites

A connection to the console port.

Procedure

1. Switch to the management interface context with the command `interface mgmt`.
2. By default, the management interface on the management port is enabled. If it was disabled, re-enable it with the command `no shutdown`.
3. Use the command `ip dhcp` to configure the management interface to automatically obtain an address from a DHCP server on the network (factory default setting). Or, assign a static IPv4 or IPv6 address, default gateway, and DNS server with the commands `ip address`, `ipv6 address`, `ip static`, `default-gateway`, and `nameserver`.
4. SSH is enabled by default on the management VRF. If disabled, enable SSH with the command `ssh server vrf mgmt`.

Examples

This example enables the management interface with dynamic addressing using DHCP:

```
switch(config)# interface mgmt
switch(config-if-mgmt)# no shutdown
switch(config-if-mgmt)# ip dhcp
```

This example enables the management interface with static addressing creating the following configuration:

- Sets a static IPv4 address of **198.168.100.10** with a mask of **24** bits.
- Sets the default gateway to **198.168.100.200**.
- Sets the DNS server to **198.168.100.201**.

```
switch(config)# interface mgmt
switch(config-if-mgmt)# no shutdown
switch(config-if-mgmt)# ip static 198.168.100.10/24
switch(config-if-mgmt)# default-gateway 198.168.100.200
switch(config-if-mgmt)# nameserver 198.168.100.201
```

Restoring the switch to factory default settings

Prerequisites

You are connected to the switch through its Console port.

This procedure erases all user information and configuration settings Consider backing up your running configuration first.

1. Optionally, back up the running configuration with either `copy running-config <REMOTE-URL>` or `copy running-config <STORAGE-URL>`. The `json` storage format is required for later configuration restoration.
2. Switch to the configuration context with the command `config`.
3. Erase all user information and configuration, restoring the switch to its factory default state with the command `erase all zeroize`. Enter `Y` when prompted to continue. The switch automatically restarts.
4. Optionally restore your saved configuration (it must be in `json` format) with either `copy <REMOTE-URL> running-config` or `copy <STORAGE-URL> running-config` followed by `copy running-config startup-config`.

Example

Backing up the running configuration to a file on a remote server (using TFTP), resetting the switch to its factory default state, and then restoring the saved configuration.

```
switch# copy running-config tftp://192.168.1.10/backup_cfg json vrf mgmt

  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 10340    0     0  100 10340      0  1329k --:--:-- --:--:-- --:--:-- 1329k
100 10340    0     0  100 10340      0  1313k --:--:-- --:--:-- --:--:-- 1313k
switch#
switch#
switch# erase all zeroize
This will securely erase all customer data and reset the switch
to factory defaults. This will initiate a reboot and render the
switch unavailable until the zeroization is complete.
This should take several minutes to one hour to complete.
Continue (y/n)? y
The system is going down for zeroization.
[  OK  ] Stopped PSPO Module Daemon.
[  OK  ] Stopped ArubaOS-CX Switch Daemon for BCM.
...
[  OK  ] Stopped Remount Root and Kernel File Systems.
[  OK  ] Reached target Shutdown.
reboot: Restarting system
Press Esc for boot options
ServiceOS Information:
```

```
    Version:          GT.01.03.0006
    Build Date:       2018-10-30 14:20:44 PDT
    Build ID:         ServiceOS:GT.01.03.0006:8ee0faaa52da:201810301420
    SHA:              xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
...

################ Preparing for zeroization #################

################ Storage zeroization ######################
################ WARNING: DO NOT POWER OFF UNTIL  ##########
################          ZEROIZATION IS COMPLETE ##########
################ This should take several minutes ##########
################ to one hour to complete          ##########

################ Restoring files ##########################

Boot Profiles:

0. Service OS Console
1. Primary Software Image [XL.10.02.0010]
2. Secondary Software Image [XL.10.02.0010]

Select profile(primary):

Booting primary software image...
Verifying Image...

Image Info:
        Name: ArubaOS-CX
     Version: XL.10.02.0010
    Build Id: ArubaOS-CX:XL.10.02.0010:feaf5b9b7f09:201901292014
  Build Date: 2019-01-29 12:43:50 PST

Extracting Image...
Loading Image...
Done.
kexec_core: Starting new kernel
System is initializing
fips_post_check[5473]: FIPS_POST: Cryptographic selftest started...SUCCESS
[  OK  ] Started Login banner readiness check.
...
8400X login: admin
Password:

switch#
switch#
switch# copy tftp://192.168.1.10/backup_cfg running-config json vrf mgmt
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100 10340  100 10340    0     0  2858k      0 --:--:-- --:--:-- --:--:-- 2858k
100 10340  100 10340    0     0  2804k      0 --:--:-- --:--:-- --:--:-- 2804k
Large configuration changes will take time to process, please be patient.
switch#
switch#
switch# copy running-config startup-config
Large configuration changes will take time to process, please be patient.
switch#
```

Management interface commands

default-gateway

Syntax

```
default-gateway <IP-ADDR>

no default-gateway <IP-ADDR>
```

Description

Assigns an IPv4 or IPv6 default gateway to the management interface. An IPv4 default gateway can only be configured if a static IPv4 address was assigned to the management interface. An IPv6 default gateway can only be configured if a static IPv6 address was assigned to the management interface. The default gateway should be on the same network segment.

The `no` form of this command removes the default gateway from the management interface.

Command context

```
config-if-mgmt
```

Parameters

<IP-ADDR>

Specifies an IP address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255, or IPv6 format (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F.

Authority

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

Examples

Setting a default gateway with the IPv4 address of **198.168.5.1**:

```
switch(config)# interface mgmt
switch(config-if-mgmt)# default-gateway 198.168.5.1
```

Setting an IPv6 address of **2001:DB8::1**:

```
switch(config)# interface mgmt
switch(config-if-mgmt)# default-gateway 2001:DB8::1
```

ip static

Syntax

```
ip static <IP-ADDR>/<MASK>

no ip static <IP-ADDR>/<MASK>
```

Description

Assigns an IPv4 or IPv6 address to the management interface.

The `no` form of this command removes the IP address from the management interface and sets the interface to operate as a DHCP client.

Command context

```
config-if-mgmt
```

Parameters

<IP-ADDR>

Specifies an IP address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255, or IPv6 format (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F.

<MASK>

Specifies the number of bits in an IPv4 or IPv6 address mask in CIDR format (`x`), where `x` is a decimal number from 0 to 32 for IPv4, and 0 to 128 for IPv6.

Authority

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

Examples

Setting an IPv4 address of **198.51.100.1** with a mask of **24** bits:

```
switch(config)# interface mgmt
switch(config-if-mgmt)# ip static 198.51.100.1/24
```

Setting an IPv6 address of **2001:DB8::1** with a mask of **32** bits:

```
switch(config)# interface mgmt
switch(config-if-mgmt)# ip static 2001:DB8::1/32
```

nameserver

Syntax

```
nameserver <PRIMARY-IP-ADDR> [ <SECONDARY-IP-ADDR> ]

no nameserver <PRIMARY-IP-ADDR> [ <SECONDARY-IP-ADDR> ]
```

Description

Assigns a primary or secondary IPv4 or IPv6 DNS server to the management interface. IPv4 DNS servers can only be configured if a static IPv4 address was assigned to the management interface. IPv6 DNS servers can only be configured if a static IPv6 address was assigned to the management interface. The default gateway should be on the same network segment.

The `no` form of this command removes the DNS servers from the management interface.

Command context

`config-if-mgmt`

Parameters

<PRIMARY-IP-ADDR>

Specifies the IP address of the primary DNS server. Specify the address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255, or IPv6 format (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F.

<SECONDARY-IP-ADDR>

Specifies the IP address of the secondary DNS server. Specify the address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255, or IPv6 format (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F.

Authority

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

Examples

Setting primary and secondary DNS servers with the IPv4 addresses of **198.168.5.1** and **198.168.5.2** :

```
switch(config)# interface mgmt
switch(config-if-mgmt)# nameserver 198.168.5.1 198.168.5.2
```

Setting primary and secondary DNS servers with the IPv6 addresses of **2001:DB8::1** and **2001:DB8::2**:

```
switch(config)# interface mgmt
switch(config-if-mgmt)# nameserver 2001:DB8::1 2001:DB8::2
```

show interface mgmt

Syntax

```
show interface mgmt [vsx-peer]
```

Description

Shows status and configuration information for the management interface.

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

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# show interface mgmt

  Address Mode                  : static
  Admin State                   : up
  Mac Address                   : 02:42:ac:11:00:02
  IPv4 address/subnet-mask      : 192.168.1.10/16
  Default gateway IPv4          : 192.168.1.1
  IPv6 address/prefix           : 2001:db8:0:1::129/64
  IPv6 link local address/prefix: fe80::7272:cfff:fefd:e485/64
  Default gateway IPv6          : 2001:db8:0:1::1
  Primary Nameserver            : 2001::1
  Secondary Nameserver          : 2001::2
```

NTP commands

ntp authentication

Syntax

```
ntp authentication

no ntp authentication
```

Description

Enables support for authentication when communicating with an NTP server.

The `no` form of this command disables authentication support.

Command context

```
config
```

Authority

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

Examples

Enabling authentication support:

```
switch(config)# ntp authentication
```

Disabling authentication support:

```
switch(config)# no ntp authentication
```

ntp authentication-key

Syntax

```
ntp authentication-key <KEY-ID> {md5 | sha1}
    [{ <PLAINTXT-KEY> [trusted] | ciphertext <ENCRYPTED-KEY> }]
no ntp authentication-key <KEY-ID>
```

Description

Defines an authentication key that is used to secure the exchange with an NTP time server. This command provides protection against accidentally synchronizing to a time source that is not trusted.

The `no` form of this command removes the authentication key.

Command context

```
config
```

Parameters

<KEY-ID>

Specifies the authentication key ID. Range: 1 to 65534.

`md5`

Selects MD5 key encryption.

`sha1`

Specifies SHA1 key encryption.

<PLAINTXT-KEY>

Specifies the plaintext authentication key. Range: 8 to 40 characters. The key may contain printable ASCII characters excluding "#" or be entered in hex. Keys longer than 20 characters are assumed to be hex. To use an ASCII key longer than 20 characters, convert it to hex.

`trusted`

Specifies that this is a trusted key. When NTP authentication is enabled, the switch only synchronizes with time servers that transmit packets containing a trusted key.

`ciphertext <ENCRYPTED-KEY>`

Specifies the ciphertext authentication key in Base64 format. This is used to restore the NTP authentication key when copying configuration files between switches or when uploading a previously saved configuration.

When the key is not provided on the command line, plaintext key prompting occurs upon pressing Enter, followed by prompting as to whether the key is to be trusted. The entered key characters are masked with asterisks.

Authority

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

Examples

Defining key 10 with MD5 encryption and a provided plaintext trusted key:

```
switch(config)# ntp authentication-key 10 md5 F82#450b trusted
```

Defining key 5 with SHA1 encryption and a prompted plaintext trusted key:

```
switch(config)# ntp authentication-key 5 sha1
Enter the NTP authentication key: *********
Re-Enter the NTP authentication key: *********

Configure the key as trusted (y/n)? y
```

Removing key 10:

```
switch(config)# no ntp authentication-key 10
```

ntp disable

Syntax

```
ntp disable
```

Description

Disables the NTP client on the switch. The NTP client is disabled by default.

Command context

```
config
```

Authority

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

Examples

Disabling the NTP client.

```
switch(config)# ntp disable
```

ntp enable

Syntax

```
ntp enable
```

```
no ntp enable
```

Description

Enables the NTP client on the switch to automatically adjust the local time and date on the switch. The NTP client is disabled by default.

The `no` form of this command disables the NTP client.

Command context

```
config
```

Authority

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

Examples

Enabling the NTP client.

```
switch(config)# ntp enable
```

Disabling the NTP client.

```
switch(config)# no ntp enable
```

ntp master

Syntax

```
ntp master vrf <VRF-NAME> {stratum <NUMBER>]
```

```
no ntp master vrf <VRF-NAME>
```

Description

Sets the switch as the master time source for NTP clients on the specified VRF. By default, the switch operates at stratum level 8. The switch cannot function as both NTP master and client on the same VRF.

The `no` form of this command stops the switch from operating as the master time source on the specified VRF.

Command context

```
config
```

Parameters

`vrf <VRF-NAME>`

Specifies the VRF on which to act as master time source.

`stratum <NUMBER>`

Specifies the stratum level at which the switch operates. Range: 1 - 15. Default: 8.

Authority

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

Examples

Setting the switch to act as master time source on VRF **primary-vrf** with a stratum level of **9**.

```
switch(config)# ntp master vrf primary-vry statum 9
```

Stops the switch from acting as master time source on VRF **primary-vrf**.

```
switch(config)# no ntp master vrf primary-vry
```

ntp server

Syntax

```
ntp server <IP-ADDR> [key <KEY-NUM>] [minpoll <MIN-NUM>] [maxpoll <MAX-NUM>][burst | iburst]
      [prefer] [version <VER-NUM>]
no ntp server <IP-ADDR>
no ntp server <IP-ADDR> [burst] [iburst] [prefer] [key-id <KEY-NUM>]
```

Description

Defines an NTP server to use for time synchronization, or updates the settings of an existing server with new values. Up to eight servers can be defined.

The `no` form of this command removes a configured NTP server.

The default NTP version is 4; it is backwards compatible with version 3.

Command context

`config`

Parameters

`server<IP-ADDR>`

Specifies the address of an NTP server as a DNS name, an IPv4 address (`x.x.x.x`), where `x` is a decimal number from 0 to 255, or an IPv6 address (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F.

When specifying an IPv4 address, you can remove leading zeros. For example, the address `192.169.005.100` becomes `192.168.5.100`.

When specifying an IPv6 address, you can use two colons (::) to represent consecutive zeros (but only once), remove leading zeros, and collapse a hextet of four zeros to a single 0. For example, this address `2222:0000:3333:0000:0000:0000:4444:0055` becomes `2222:0:3333::4444:55`.

`key <KEY-NUM>`

Specifies the key to use when communicating with the server. A trusted key must be defined with the command `ntp authentication-key` and authentication must be enabled with the command `ntp authentication`. Range: 1 to 65534.

`minpoll <MIN-NUM>`

Specifies the minimum polling interval in seconds, as a power of 2. Range: 4 to 17. Default: 6 (64 seconds).

`maxpoll <MAX-NUM>`

Specifies the maximum polling interval in seconds, as a power of 2. Range: 4 to 17. Default: 10 (1024 seconds).

`burst`

Send a burst of packets instead of just one when connected to the server. Useful for reducing phase noise when the polling interval is long.

`iburst`

Send a burst of six packets when not connected to the server. Useful for reducing synchronization time at startup.

`prefer`

Make this the preferred server.

`version <VER-NUM>`

Specifies the version number to use for all outgoing NTP packets. Range: 3 or 4.

Authority

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

Usage

For features such as Activate and ZTP, a switch that has a factory default configuration will automatically be configured with pool.ntp.org. NTP server configurations via DHCP options are supported. The DHCP server can be configured with maximum of two NTP server addresses which will be supported on the switch. Only IPV4 addresses are supported.

Examples

Defining the ntp server pool.ntp.org, using iburst, and NTP version 4.

```
switch(config)# ntp server pool.ntp.org iburst version 4
```

Removing the ntp server pool.ntp.org.

```
switch(config)# no ntp server pool.ntp.org
```

Defining the ntp server my-ntp.mydomain.com and makes it the preferred server.

```
switch(config)# ntp server my-ntp.mydomain.com prefer
```

ntp trusted-key

Syntax

`ntp trusted-key <KEY-ID>`

`no ntp trusted-key <KEY-ID>`

Description

Sets a key as trusted. When NTP authentication is enabled, the switch only synchronizes with time servers that transmit packets containing a trusted key.

The `no` form of this command removes the trusted designation from a key.

Command context

`config`

Parameters

`<KEY-ID>`

Specifies the identification number of the key to set as trusted. Range: 1 to 65534.

Authority

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

Examples

Defining key 10 as a trusted key.

```
switch(config)# ntp trusted-key 10
```

Removing trusted designation from key 10:

```
switch(config)# no ntp trusted-key 10
```

ntp vrf

Syntax

`ntp vrf <VRF-NAME>`

Description

Specifies the VRF on which the NTP client communicates with an NTP server. The switch cannot function as both NTP master and client on the same VRF.

Command context

`config`

Parameters

`<VRF-NAME>`

Specifies the name of a VRF.

Authority

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

Example

Setting the switch to use the default VRF for NTP client traffic.

```
switch(config)# ntp vrf default
```

Setting the switch to use the default management VRF for NTP client traffic.

```
switch(config)# ntp vrf mgmt
```

show ntp associations

Syntax

```
show ntp associations [vsx-peer]
```

Description

Shows the status of the connection to each NTP server. The following information is displayed for each server:

- Tally code : The first character is the Tally code:
  - (blank): No state information available (e.g. non-responding server)
  - x : Out of tolerance (discarded by intersection algorithm)
  - . : Discarded by table overflow (not used)
  - - : Out of tolerance (discarded by the cluster algorithm)
  - + : Good and a preferred remote peer or server (included by the combine algorithm)
  - # : Good remote peer or server, but not utilized (ready as a backup source)
  - * : Remote peer or server presently used as a primary reference
  - o : PPS peer (when the prefer peer is valid)
- ID: Server number.
- NAME: NTP server FQDN/IP address (Only the first 24 characters of the name are displayed).
- REMOTE: Remote server IP address.
- REF_ID: Reference ID for the remote server (Can be an IP address).
- ST: (Stratum) Number of hops between the NTP client and the reference clock.
- LAST: Time since the last packet was received in seconds unless another unit is indicated.
- POLL: Interval (in seconds) between NTP poll packets. Maximum (1024) reached as server and client sync.
- REACH: 8-bit octal number that displays status of the last eight NTP messages (377 = all messages received).

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

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# show ntp associations
-------------------------------------------------------------------------
 ID            NAME          REMOTE          REF-ID ST LAST POLL REACH
-------------------------------------------------------------------------
  1       192.0.1.1       192.0.1.1          .INIT. 16    -   64     0
* 2  time.apple.com    17.253.2.253          .GPSs.  2   70  128   377
-------------------------------------------------------------------------
```

show ntp authentication-keys

Syntax

```
show ntp authentication-keys [vsx-peer]
```

Description

Shows the currently defined authentication keys.

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

```
switch# show ntp authentication-keys
--------------------------------
Auth key  Trusted   MD5 password
--------------------------------
 10          No     **********
 20         Yes     **********
```

show ntp servers

Syntax

```
show ntp servers[vsx-peer]
```

Description

Shows all configured NTP servers.

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

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# show ntp servers
-------------------------------------------------
     NTP SERVER KEYID MINPOLL MAXPOLL OPTION VER
-------------------------------------------------
     192.0.1.18     -       5      10 iburst   3
     192.0.1.19     -       6      10   none   4
     192.0.1.20     -       6       8  burst   3 prefer
-------------------------------------------------
```

show ntp statistics

Syntax

```
show ntp statistics [vsx-peer]
```

Description

Shows global NTP statistics. The following information is displayed:

- Rx-pkts: Total NTP packets received.
- Current Version Rx-pkts: Number of NTP packets that match the current NTP version.
- Old Version Rx-pkts: Number of NTP packets that match the previous NTP version.
- Error pkts: Packets dropped due to all other error reasons.
- Auth-failed pkts: Packets dropped due to authentication failure.
- Declined pkts: Packets denied access for any reason.
- Restricted pkts: Packets dropped due to NTP access control.
- Rate-limited pkts: Number of packets discarded due to rate limitation.
- KOD pkts: Number of Kiss of Death packets sent.

Command context

```
config
```

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

```
switch(config)# show ntp statistics
                 Rx-pkts 100
Current Version Rx-pkts 80
    Old Version Rx-pkts 20
               Err-pkts 2
       Auth-failed-pkts 1
          Declined-pkts 0
        Restricted-pkts 0
      Rate-limited-pkts 0
               KoD-pkts 0
```

show ntp status

Syntax

```
show ntp status [vsx-peer]
```

Description

Shows the status of NTP on the switch.

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

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

Displaying the status information when the switch is not synced to an NTP server:

```
switch# show ntp status
NTP is enabled.
NTP authentication is enabled.
NTP is using the default VRF for NTP server connections.

Wed Nov 23 23:29:10 PDT 2016
NTP uptime: 187 days, 1 hours, 37 minutes, 48 seconds

Not synchronized with an NTP server.
```

Displaying the status information when the switch is synced to an NTP server:

```
switch# show ntp status
NTP is enabled.
NTP authentication is enabled.
```

```
NTP is using the default VRF for NTP server connections.

Wed Nov 23 23:29:10 PDT 2016
NTP uptime: 187 days, 1 hours, 37 minutes, 48 seconds

Synchronized to NTP Server 17.253.2.253 at stratum 2.
Poll interval = 1024 seconds.
Time accuracy is within 0.994 seconds
Reference time: Thu Jan 28 2016 0:57:06.647 (UTC)
```

Configuring a layer 2 interface

Procedure

1. Change to the interface configuration context for the interface with the command `interface`.
2. Set the interface MTU (maximum transmission unit) with the command `mtu`.
3. Review interface configuration settings with the command `show interface`.

Example

On the 6300 switch series:

```
switch(config)# interface 1/1/1
switch(config-if)# mtu 1900
```

On the 6400 switch series:

```
switch(config)# interface 1/3/1
switch(config-if)# mtu 1900
```

Configuring a layer 3 interface

Procedure

1. Change to the interface configuration context for the interface with the command `interface`.
2. Enable routing support with the command `routing`.
3. Assign an IPv4 address with the command `ip address`, or an IPv6 address with the command `ipv6 address`.
4. If required, enable support for layer 3 counters with the command `l3-counters`.
5. If required, set the IP MTU with the command `ip mtu`.
6. Review interface configuration settings with the command `show interface`.

Examples

This example creates the following configuration on the 6300 Switch Series:

- Configures interface **1/1/1** as a layer 3 interface.
- Defines an IPv4 address of **10.10.20.209** with a 24-bit mask.

```
switch# config
switch(config)# interface 1/1/1
```

```
switch(config-if)# routing
switch(config-if)# ip address 10.10.20.209/24
```

This example creates the following configuration on the 6400 Switch Series:

- Configures interface **1/3/1** as a layer 3 interface.
- Defines an IPv6 address of **2001:0db8:85a3::8a2e:0370:7334** with a 24-bit mask.
- Enables layer 3 transmit and receive counters.

```
switch# config
switch(config)# interface 1/3/1
switch(config-if)# routing
switch(config-if)# ipv6 address 2001:0db8:85a3::8a2e:0370:7334/24
switch(config-if)# l3-counters tx
switch(config-if)# l3-counters rx
```

Single source IP address

Certain IP-based protocols used by the switch (such as RADIUS, sFlow, TACACS, and TFTP), use a client-server model in which the client's source IP address uniquely identifies the client in packets sent to the server. By default, the source IP address is defined as the IP address of the outgoing switch interface on which the client is communicating with the server. Since the switch can have multiple routing interfaces, outgoing packets can potentially be sent on different paths at different times. This can result in different source IP addresses being used for a client, which can create a client identification problem on the server. For example, it can be difficult to interpret system logs and accounting data on the server when the same client is associated with multiple IP addresses.

To resolve this issue, you can use the commands `ip source-interface` and `ipv6 source-interface` to define a single source IP address that applies to all supported protocols (RADIUS, sFlow, TACACS, and TFTP), or an individual address for each protocol. This ensures that all traffic sent by a client to a server uses the same IP address.

Unsupported transceiver support

Transceiver products (optical, DAC, AOCs) that are listed as supported by a switch model are detailed in the *Transceiver Guide*. Transceiver products that are not listed, are considered unsupported; this would include transceivers that are:

- Non-Aruba branded products
- HPE branded products that were designed for non-AOS-CX switch models (e.g. Comware)
- HPE branded products designated for use in HPE Compute Servers or Storage
- Transceivers originally designated for use in Aruba WLAN controllers or former Mobility Access Switch (MAS) products
- End-of-life Aruba Transceivers

The unsupported transceiver mode (UT-mode) is designed to allow the possible use of these unsupported products. Not all unsupported products can be recognized and enabled; they may be unable to be identified (do not follow the proper MSA standards for identification). These unsupported transceiver products are enabled only on a best-effort basis and there are no guarantees implied for their continued operation.

The feature is disabled by default. A periodic system log will be generated by default at an interval of 24 hours listing the ports on which unsupported transceivers are present. The log interval is configurable and can be disabled by setting the log-interval to `none`.

Interface commands

allow-unsupported-transceiver

Syntax

```
allow-unsupported-transceiver [confirm | log-interval {none | <INTERVAL>}]

no allow-unsupported-transceiver
```

Description

Allows unsupported transceivers to be enabled or establish connections. Only 1G and 10G transceivers are enabled by this command and unsupported transceivers of other speeds will remain disabled.

The `no` form of this command disallows using unsupported transceivers. This is the default.

Command context

`config`

Parameters

`confirm`

Specifies that unsupported transceiver warnings are to be automatically confirmed.

`log-interval none`

Disables unsupported transceiver logging.

`log-interval <INTERVAL>`

Sets the unsupported transceiver logging interval in minutes. Default: 1440 minutes. Range: 1440 to 10080 minutes.

Authority

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

Usage

When none of the parameters are specified it will display a warning message to accept the warranty terms. With `confirm` option the warning message is displayed but the user is not prompted to `(y/n)` answering. Warranty terms must be agreed to as part of enablement and the support is on best effort basis.

Examples

Allowing unsupported transceivers with follow-up confirmation:

```
switch(config)# allow-unsupported-transceiver
Warning: The use of unsupported transceivers, DACs, and AOCs is at your
own risk and may void support and warranty. Please see HPE Warranty terms
and conditions.

Do you agree and do you want to continue (y/n)? y
```

Allowing unsupported transceivers with confirmation in command syntax:

```
switch(config)# allow-unsupported-transceiver confirm
Warning: The use of unsupported transceivers, DACs, and AOCs is at your
own risk and may void support and warranty. Please see HPE Warranty terms
and conditions.
```

Configuring unsupported transceiver logging with an interval of every 48 hours:

```
switch(config)# allow-unsupported-transceiver log-interval 2880
```

Disabling unsupported transceiver logging:

```
switch(config)# allow-unsupported-transceiver log-interval none
```

Disallowing unsupported transceivers with follow-up confirmation:

```
switch(config)# no allow-unsupported-transceiver
Warning: Unsupported transceivers, DACs, and AOCs will be disabled,
which could impact network connectivity. Use 'show allow-unsupported-transceiver'
to identify unsupported transceivers, DACs, and AOCs.

Continue (y/n)? y
```

Disallowing unsupported transceivers with confirmation in command syntax:

```
switch(config)# no allow-unsupported-transceiver confirm
Warning: Unsupported transceivers, DACs, and AOCs will be disabled,
which could impact network connectivity. Use 'show allow unsupported-transceiver'
to identify unsupported transceivers, DACs, and AOCs.

switch(config)#
```

default interface

Syntax

`default interface <INTERFACE-ID>`

Description

Sets an interface (or a range of interfaces) to factory default values.

Command context

`config`

Parameters

<INTERFACE-ID>

Specifies the ID of a single interface or range of interfaces. Format: `member/slot/port` or `member/slot/port-member/slot/port` to specify a range.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Resetting an interface:

```
switch(config)# default default interface 1/1/1
```

Resetting an range of interfaces:

```
switch(config)# default default interface 1/1/1-1/1/10
```

description

Syntax

```
description <DESCRIPTION>
no description
```

Description

Associates descriptive information with an interface to help administrators and operators identify the purpose or role of an interface.

The `no` form of this command removes a description from an interface.

Command context

`config-if`

Parameters

<DESCRIPTION>

Specify a description for the interface. Range: 1 to 64 ASCII characters (including space, excluding question mark).

Authority

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

Examples

Setting the description for an interface to **DataLink 01**:

```
switch(config-if)# description DataLink 01
```

Removing the description for an interface.

```
switch(config-if)# no description
```

energy-efficient-ethernet

Syntax

```
energy-efficient-ethernet
```

Description

Enables auto-negotiation of Energy-Efficient Ethernet (EEE) on an interface. EEE Negotiation is established only on auto-link negotiation with supported link partners.

Command context

`config`

Authority

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

Examples

Configuring an interface:

```
switch(config)# interface 1/1/1
switch(config-if)# energy-efficient-ethernet
```

Disabling Energy Efficient Ethernet on an interface:

```
switch(config)# interface 1/1/1
switch(config-if)# no energy-efficient-ethernet
```

flow-control

Syntax

```
flow-control rxtx

[no] flow-control rxtx
```

Description

Enables negotiation of IEEE 802.3x link-level flow control on the current interface. The switch advertises link-level flow-control support to the link partner. The final configuration is determined based on the capabilities of both partners.

The no form disables flow control support on the current interface.

Command context

`config-if`

Authority

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

Parameters

`rxtx`

Enables the ability to respect and generate IEEE 802.3x link-level pause frames on the current interface.

Examples

Enable support for rxtx flow control:

```
switch(config)# interface 1/1/1
switch(config-if)# flow-control rxtx
```

Disable support for rxtx flow control:

```
switch(config)# interface 1/1/1
switch(config-if)# no flow-control rxtx
```

interface

Syntax

`interface <PORT-NUM>`

Description

Switches to the `config-if` context for a physical port. This is where you define the configuration settings for the logical interface associated with the physical port.

Command context

`config`

Parameters

<PORT-NUM>

Specifies a physical port number. Format: `member/slot/port`.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Configuring an interface:

```
switch(config)# interface 1/1/1
switch(config-if)#
```

interface loopback

Syntax

`interface loopback <ID>`

`no interface loopback <ID>`

Description

Creates a loopback interface and changes to the `config-loopback-if` context. Loopback interfaces are layer 3.

The `no` form of this command deletes a loopback interface.

Command context

`config`

Parameters

<INSTANCE>

Specifies the loopback interface ID. Range: 1 to 256

Authority

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

Examples

```
switch# config
switch(config)# interface loopback 1
switch(config-loopback-if)#
```

interface vlan

Syntax

`interface vlan <VLAN-ID>`

`no interface vlan <VLAN-ID>`

Description

Creates an interface VLAN also know as an SVI (switched virtual interface) and changes to the `config-if-vlan` context. The specified VLAN must already be defined on the switch.

The `no` form of this command deletes an interface VLAN.

Command context

`config`

Parameters

<VLAN-ID>

Specifies the loopback interface ID. Range: 2 to 4094

Authority

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

Examples

```
switch# config
switch(config)# vlan 10
switch(config-vlan-10)# exit
switch(config)# interface vlan 10
switch(config-if-vlan)#
```

ip address

Syntax

`ip address <IPV4-ADDR>/<MASK> [secondary]`

`no ip address <IPV4-ADDR>/<MASK> [secondary]`

Description

Sets an IPv4 address for the current layer 3 interface.

The `no` form of this command removes the IPv4 address from the interface.

Command context

```
config-if
config-loopback-if
config-if-vlan
```

Parameters

<IPV4-ADDR>

Specifies an IP address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255. You can remove leading zeros. For example, the address `192.169.005.100` becomes `192.168.5.100`.

<MASK>

Specifies the number of bits in the address mask in CIDR format (`x`), where `x` is a decimal number from 0 to 128.

`secondary`

Specifies a secondary IP address.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Creating a layer 3 interface setting its IP address to **192.168.100.1** with a mask of **24** bits.

```
switch(config)# interface 1/1/1
switch(config-if)# routing
switch(config-if)# ip address 192.168.100.1/24
```

Assigning the IP address **192.168.20.1** with a mask of **24** bits to loopback interface **1**:

```
switch(config)# interface loopback 1
switch(config-loopback-if)# routing
switch(config-loopback-if)# ip address 192.168.20.1/24
```

Assigning the IP address **192.168.199.1** with a mask of **24** bits to interface VLAN **10**:

```
switch(config)# interface vlan 10
switch(config-if-vlan)# ip address 192.168.199.1/24
```

Removing the IP address **192.168.199.1** with a mask of **24** bits from interface VLAN **10**:

```
switch(config)# interface vlan 10
switch(config-if-vlan)# no ip address 192.168.199.1/24
```

ip mtu

Syntax

```
ip mtu <VALUE>

ip no mtu
```

Description

Sets the IP MTU (maximum transmission unit) for an interface. This defines the largest IP packet that can be sent or received by the interface.

The `no` form of this command sets the IP MTU to the default value 1500.

Command context

```
config-if
config-if-vlan
```

Parameters

<VALUE>

Specifies the IP MTU in bytes. Range: 68 to 9198. Default: 1500.

Authority

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

Examples

Setting the IP MTU to 576 bytes:

```
switch(config-if)# ip mtu 576
```

Setting the IP MTU to the default value:

```
switch(config-if)# no ip mtu
```

ip source-interface

Syntax

```
ip source-interface {sflow |  tftp |  radius | tacacs | ntp | syslog | ubt | dhcp-relay |
 simplivity | dns | all} {interface <IFNAME> | <IPV4-ADDR>} [vrf <VRF-NAME>]

no ip source-interface {sflow |  tftp |  radius | tacacs | ntp | syslog | ubt | dhcp-relay |
 simplivity | dns | all} [interface <IFNAME> | <IPV4-ADDR>] [vrf <VRF-NAME>]
```

Description

Sets a single source IP address for a feature on the switch. This ensures that all traffic sent the feature has the same source IP address regardless of how it egresses the switch. You can define a single global address that applies to all supported features, or an individual address for each feature.

This command provides two ways to set the source IP addresses: either by specifying a static IP address, or by using the address assigned to a switch interface. If you define both options, then the static IP address takes precedence.

The `no` form of this command deletes the single source IP address for all supported services, or a specific service.

Command context

```
config
```

Parameters

```
sflow | tftp | radius | tacacs | ntp | syslog |  ubt | dhcp-relay | simplivity | dns | all
```

Sets a single source IP address for a specific service. The `all` option sets a global address that applies to all protocols that do not have an address set. For DHCP relay, the address is used as both the source IP and GIADDR.

`interface <IFNAME>`

Specifies the name of the interface from which the specified service obtains its source IP address. The interface must have a valid IP address assigned to it. If the interface has both a primary and secondary IP address, the primary IP address is used.

`<IPV4-ADDR>`

Specifies the source IP address to use for the specified service. The IP address must be defined on the switch, and it must exist on the specified VRF (which is the `default` VRF, if the `vrf` option is not used). Specify the address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255.

`vrf <VRF-NAME>`

Specifies the name of a VRF.

Authority

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

Examples

Setting the IPv4 address `10.10.10.5` as the global single source address:

```
switch# config
switch(config)# ip source-interface all 10.10.10.5
```

Setting the secondary IPv4 address `10.10.10.5` on interface `1/1/1` as the global single source address. (On the 6400 Switch Series, interface identification differs.)

```
switch# config
switch(config)# interface 1/1/1
switch(config-if)# routing
switch(config-if)# ip address 10.10.10.1/24
switch(config-if)# ip address 10.10.10.5/24 secondary
switch(config)# exit
switch(config)# ip source-interface all 10.10.10.5
```

Clearing the global single source IP address **10.10.10.5**:

```
switch(config)# no ip source-interface all 10.10.10.5
```

ipv6 address

Syntax

`ipv6 address <IPV6-ADDR>/<MASK>{eui64 | [tag <ID>]}`

`no ipv6 address <IPV6-ADDR>/<MASK>`

Description

Sets an IPv6 address on the interface.

The `no` form of this command removes the IPv6 address on the interface.

This command automatically creates an IPv6 link-local address on the interface. However, it does not add the `ipv6 address link-local` command to the running configuration. If you remove the IPv6 address, the link-local address is also removed. To maintain the link-local address, you must manually execute the `ipv6 address link-local` command.

Command context

`config-if`

Parameters

<IPV6-ADDR>

Specifies the IP address in IPv6 format (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F. You can use two colons (::) to represent consecutive zeros (but only once), remove leading zeros, and collapse a hextet of four zeros to a single 0. For example, this address `2222:0000:3333:0000:0000:0000:4444:0055` becomes `2222:0:3333::4444:55`.

<MASK>

Specifies the number of bits in the address mask in CIDR format (`x`), where `x` is a decimal number from 0 to 128.

`eui64`

Configure the IPv6 address in the EUI-64 bit format.

`tag` *<ID>*

Configure route tag for connected routes. Range: 0 to 4294967295. Default: 0.

Authority

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

Examples

Setting the IPv6 address **2001:0db8:85a3::8a2e:0370:7334** with a mask of 24 bits:

```
switch(config-if)# ipv6 address 2001:0db8:85a3::8a2e:0370:7334/24
```

Removing the IP address 2001:0db8:85a3::8a2e:0370:7334 with mask of 24 bits:

```
switch(config-if)# no ipv6 address 2001:0db8:85a3::8a2e:0370:7334/24
```

ipv6 source-interface

Syntax

```
ipv6 source-interface {sflow |  tftp |  radius | tacacs | ntp | syslog | ubt | dhcp-relay |
simplivity | dns | all} {interface <IFNAME> | <IPV6-ADDR>} [vrf <VRF-NAME>]

no ipv6 source-interface {sflow |  tftp |  radius | tacacs | ntp | syslog | ubt | dhcp-relay
| simplivity | dns | all} [interface <IFNAME> | <IPV6-ADDR>] [vrf <VRF-NAME>]
```

Description

Sets a single source IP address for a feature on the switch. This ensures that all traffic sent the feature has the same source IP address regardless of how it egresses the switch. You can define a single global address that applies to all supported features, or an individual address for each feature.

This command provides two ways to set the source IP addresses: either by specifying a static IP address, or by using the address assigned to a switch interface. If you define both options, then the static IP address takes precedence.

The `no` form of this command deletes the single source IP address for all supported protocols, or a specific protocol.

Command context

`config`

Parameters

`sflow | tftp | radius | tacacs | ntp | syslog | ubt | dhcp-relay | simplivity | dns | all`

Sets a single source IP address for a specific protocol. The `all` option sets a global address that applies to all protocols that do not have an address set.

`interface <IFNAME>`

Specifies the name of the interface from which the specified protocol obtains its source IP address.

`<IPV6-ADDR>`

Specifies the source IP address to use for the specified protocol. The IP address must be defined on the switch, and it must exist on the specified VRF (which is the `default` VRF, if the `vrf` option is not used). Specify the IP address in IPv6 format (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F.

`vrf <VRF-NAME>`

Specifies the name of the VRF from which the specified protocol sets its source IP address.

Authority

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

Examples

Configuring the IPv6 address `2001:DB8::1` as the global single source address:

```
switch# config
switch(config)# ip source-interface all 2001:DB8::1/32
```

Configuring the IPv6 address `2001:DB8::1` on VRF `sflow-vrf` on interface `1/1/2` as the single source address for sFlow:

```
switch(config)# vrf sflow-vrf
switch(config-vrf)# exit
switch(config)# interface 1/1/2
switch(config-if)# no shutdown
switch(config-if)# vrf attach sflow-vrf
switch(config-if)# ipv6 address 2001:DB8::1/32
switch(config-if)# exit
switch(config)# ip source-interface sflow interface 1/1/2 vrf sflow-vrf
```

Stop the source IP address from using the IP address on interface **1/1/1** on VRF `one`.

```
switch(config)# no ip source-interface all interface 1/1/1 vrf one
```

Clear the source IP address `2001:DB8::1`.

```
switch(config)# no ip source-interface all 2001:DB8::1
```

l3-counters

Syntax

```
l3-counters [rx | tx]

no l3-counters [rx | tx]
```

Description

Enables counters on a layer 3 interface. By default, all interfaces are layer 3. To change a layer 2 interface to layer 3, use the `routing` command.

The `no` form of this command, with no specification, disables both transmit and receive counters on a layer 3 interface. To disable transmit (`tx`) or receive (`rx`) counters only, specify the counter type you want to disable.

Command context

```
config-if
```

Parameters

`rx`
: Specifies receive counters.

`tx`
: Specifies transmit counters.

Authority

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

Examples

Enabling layer 3 transmit counters.

On the 6300 Switch Series:

```
switch(config)# interface 1/1/1
switch(config-if)# routing
switch(config-if)# l3-counters
```

On the 6400 Switch Series:

```
switch(config)# interface 1/3/1
switch(config-if)# routing
switch(config-if)# l3-counters
```

mtu

Syntax

```
mtu <VALUE>

no mtu
```

Description

Sets the MTU (maximum transmission unit) for an interface. This defines the maximum size of a layer 2 (Ethernet) frame. Frames larger than the MTU (1500 bytes by default) are dropped and cause an ICMP fragmentation-needed message to be sent back to the originator.

To support jumbo frames (frames larger than 1522 bytes), increase the MTU as required by your network. A frame size of up to 9198 bytes is supported.

The largest possible layer 1 frame will be 18 bytes larger than the MTU value to allow for link layer headers and trailers.

The `no` form of this command sets the MTU to the default value 1500.

Command context

`config-if`

Parameters

<VALUE>

Specifies the MTU in bytes. Range: 46 to 9198. Default: 1500.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Setting the MTU on interface **1/1/1** to 1000 bytes:

```
switch(config)# interface 1/1/1
switch(config-if)# no routing
switch(config-if)# mtu 1000
```

Setting the MTU on interface **1/1/1** to the default value:

```
switch(config)# interface 1/1/1
switch(config-if)# no routing
switch(config-if)# no mtu
```

routing

Syntax

`routing`

`no routing`

Description

Enables routing support on an interface, creating a L3 (layer 3) interface on which the switch can route IPv4/IPv6 traffic to other devices.

By default, routing is disabled on all interfaces.

The `no` form of this command disables routing support on an interface, creating a L2 (layer 2) interface.

Command context

`config-if`

Authority

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

Examples

Enabling routing support on an interface:

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

Disabling routing support on an interface:

```
switch(config-if)# no routing
```

show allow-unsupported-transceiver

Syntax

```
show allow-unsupported-transceiver
```

Description

Displays configuration and status of unsupported transceivers.

Command context

```
config
```

Authority

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

Examples

Showing unallowed unsupported transceivers:

```
switch(config)# show allow-unsupported-transceiver

Allow unsupported transceivers : no
Logging interval               : 1440 minutes

-------------------------------------------
Port       Type           Status
-------------------------------------------
1/1/31     SFP-SX         unsupported
1/1/32     SFP-1G-BXD     unsupported
1/1/2      SFP28DAC3      unsupported
```

Showing allowed unsupported transceivers:

```
switch# show allow-unsupported-transceiver

Allow unsupported transceivers : yes
Logging interval               : 1440 minutes

-------------------------------------------
Port       Type           Status
```

```
-----------------------------------------------
1/1/31      SFP-SX          unsupported-allowed
1/1/32      SFP-1G-BXD      unsupported-allowed
1/1/2       SFP28DAC3       unsupported
```

show interface

Syntax

```
show interface [<IFNNAME>|<IFRANGE>] [brief | physical | extended [non-zero]]
show interface [lag | loopback | tunnel | vlan ] [<ID>] [brief | physical]
show interface [lag | loopback | tunnel | vlan ] [<ID>] [extended [non-zero]]
```

Description

Displays active configurations and operational status information for interfaces.

Command context

Operator (>) or Manager (#)

Parameters

<IFNAME>
: Specifies a interface name.

<IFRANGE>
: Specifies the port identifier range.

`brief`
: Shows brief info in tabular format.

`physical`
: Shows the physical connection info in tabular format.

`extended`
: Shows additional statistics.

`non-zero`
: Shows only non zero statistics.

`LAG`
: Shows LAG interface information.

`LOOPBACK`
: Shows loopback interface information.

`TUNNEL`
: Shows tunnel interface information.

`VLAN`
: Shows VLAN interface information.

<LAG-ID>
: Specifies the LAG number. Range: 1-256

<LOOPBACK-ID>
: Specifies the LOOPBACK number. Range: 0-255

<TUNNEL-ID>
: Specifies the tunnel ID. Range: 1-255

<VLAN-ID>
: Specifies the VLAN ID. Range: 1-4094

`VXLAN`
: Shows the VXLAN interface information.

<VXLAN-ID>

Specifies the VXLAN interface identifier. Default: 1

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

The following example shows when the interface is configured as a route-only port:

```
switch# show interface 1/1/1

Interface 1/1/1 is up
 Admin state is up
 Link state: up for 2 days (since Sun Jun 21 05:30:22 UTC 2020)
 Link transitions: 1
 Description: backup data center link
 Hardware: Ethernet, MAC Address: 70:72:cf:fd:e7:b4
 MTU 1500
 Type 1GbT
 Full-duplex
 qos trust none
 Speed 1000 Mb/s
 Auto-negotiation is on
 Flow-control: off
 Error-control: off
 Energy-Efficient Ethernet is enabled
 MDI mode: MDIX
 L3 Counters: Rx Enabled, Tx Enabled
Rate collection interval: 300 seconds

 Rates                          RX                   TX        Total (RX+TX)
 -------------- -------------------- -------------------- --------------------
 Mbits / sec                  0.00                 0.00                 0.00
 KPkts / sec                  0.00                 0.00                 0.00
   Unicast                    0.00                 0.00                 0.00
   Multicast                  0.00                 0.00                 0.00
   Broadcast                  0.00                 0.00                 0.00
 Utilization %                0.00                 0.00                 0.00

 Statistics                     RX                   TX                Total
 -------------- -------------------- -------------------- --------------------
 Packets                         0                    0                    0
   Unicast                       0                    0                    0
   Multicast                     0                    0                    0
   Broadcast                     0                    0                    0
 Bytes                           0                    0                    0
 Jumbos                          0                    0                    0
 Dropped                         0                    0                    0
 Filtered                        0                    0                    0
 Pause Frames                    0                    0                    0
 L3 Packets                      0                    0                    0
 L3 Bytes                        0                    0                    0
 Errors                          0                    0                    0
   CRC/FCS                       0                  n/a                    0
   Collision                   n/a                    0                    0
   Runts                         0                  n/a                    0
   Giants                        0                  n/a                    0
   Other                         0                    0                    0
```

When the interface is currently linked at a downshifted speed:

```
switch(config-if)# show interface 1/1/1

Interface 1/1/1 is up
 ...
 Auto-negotiation is on with downshift active
```

When the interface is currently linked with energy-efficient-ethernet negotiated:

```
switch(config-if)# show interface 1/1/1


Interface 1/1/1 is up
 ...
 Energy-Efficient Ethernet is enabled and active
```

When the interface is shut down during a VSX split:

```
switch(config-if)# show interface 1/1/1

Interface 1/1/1 is down
 Admin state is up
 State information: Disabled by VSX
 Link state: down for 3 days (since Tue Mar 16 05:20:47 UTC 2021)
 Link transitions: 0
 Description:
 Hardware: Ethernet, MAC Address: 04:09:73:62:90:e7
 MTU 1500
 Type SFP+DAC3
 Full-duplex
 qos trust none
 Speed 0 Mb/s
 Auto-negotiation is off
 Flow-control: off
 Error-control: off
 VLAN Mode: native-untagged
 Native VLAN: 1
 Allowed VLAN List: 1502-1505
 Rate collection interval: 300 seconds

 Rate                              RX                   TX        Total (RX+TX)
 ---------------- -------------------- -------------------- --------------------
 Mbits / sec                       0.00                 0.00                 0.00
 KPkts / sec                       0.00                 0.00                 0.00
   Unicast                         0.00                 0.00                 0.00
   Multicast                       0.00                 0.00                 0.00
   Broadcast                       0.00                 0.00                 0.00
 Utilization                       0.00                 0.00                 0.00

 Statistic                         RX                   TX                Total
 ---------------- -------------------- -------------------- --------------------
 Packets                            0                    0                    0
   Unicast                          0                    0                    0
   Multicast                        0                    0                    0
   Broadcast                        0                    0                    0
 Bytes                              0                    0                    0
 Jumbos                             0                    0                    0
 Dropped                            0                    0                    0
 Pause Frames                       0                    0                    0
 Errors                             0                    0                    0
```

```
   CRC/FCS                          0                 n/a                   0
   Collision                      n/a                   0                   0
   Runts                            0                 n/a                   0
   Giants                           0                 n/a                   0
```

When the interface is configured with EEE and the EEE has auto-negotiated:

```
switch(config-if)# show interface 1/1/1 physical

------------------------------------------------------------------------------------
------------------------------------------------------------------------------------
----
                              Link    Admin         Speed           Flow-Control
   EEE       PoE Power                                            Port
Port        Type            Status   Config   Status | Config    Status | Config
Status | Config  (Watts)    State Information                        Description
------------------------------------------------------------------------------------
------------------------------------------------------------------------------------
----
1/1/1       1GbT            up       up       1G       auto      off      off      on
      on       --        10M/100M/1G                              --
```

show interface dom

Syntax

```
show interface [<INTERFACE-ID>] dom [detail] [vsx-peer]
```

Description

Shows diagnostics information and alarm/warning flags for the optical transceivers (SFP, SFP+, QSFP+). This information is known as DOM (Digital Optical Monitoring). DOM information also consists of vendor determined thresholds which trigger high/low alarms and warning flags.

Command context

Manager (#)

Parameters

<INTERFACE-ID>

Specifies an interface. Format: `member/slot/port`.

`detail`

Show detailed information.

`[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

On the 6400 Switch Series, interface identification differs.

```
switch# show interface dom

--------------------------------------------------------------------------------
Port    Type      Channel  Temperature Voltage  Tx Bias  Rx Power      Tx Power
                           (Celsius)   (Volts)  (mA)     (mW/dBm)      (mW/dBm)
--------------------------------------------------------------------------------
1/1/1   SFP+SR             47.65       3.31     8.40     0.08, -10.96  0.63, -2.49
1/1/2   SFP+SR             n/a         n/a      n/a      n/a           n/a
1/1/3   SFP+DA3            42.10       3.24     n/a      n/a           n/a
1/1/4  QSFP+SR4 1    44.46    3.30   6.12   0.08, -10.96 0.63, -1.95
        2    44.46    3.30   6.04   0.08, -10.96 0.63, -2.00
        3    44.46    3.30   6.51   0.08, -10.96 0.60, -2.16
        4    44.46    3.30   6.19   0.08, -10.96 0.63, -1.94
```

show interface energy-efficient ethernet

Syntax

```
show interface [<IFNAME>|<IFRANGE>] energy-efficient-ethernet
```

Description

Displays Energy-Efficient Ethernet information for the interface.

Command context

```
config
```

Parameters

<IFNAME>

Specifies the name of an interface on the switch. Use the format `member/slot/port` (for example, `1/1/1`).

<IFRANGE>

Specifies the port identifier range of an interface on the switch. Use the format `member/slot/port` (for example, `1/1/1`).

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

The following example shows when the interfaces are Energy-Efficient Ethernet capable:

```
switch# show interface energy-efficient-ethernet
--------------------------------------------------------------------
Port     Enabled    Negotiated   Speed       TX Wake     RX Wake
                                 (MB/s)      Time(us)    Time (us)
--------------------------------------------------------------------
1/1/1    no         no           --          --         --
1/1/2    yes        yes          100         36         36
1/1/3    yes        yes          1000        17         17
1/1/4    no         no           --          --         --
1/1/5    yes        no           1000        --         --
```

The following example shows when the interface is not Energy-Efficient Ethernet capable :

```
switch# show interface 1/1/1 energy-efficient-ethernet
   Port 1/1/1 does not support Energy-Efficient-Ethernet
```

show interface transceiver

Syntax

```
show interface [<INTERFACE-ID>] transceiver [detail | threshold-violations] [vsx-peer]
```

Description

Displays information about transceivers present in the switch. The information shown varies for different transceiver types and manufacturers. Only basic information is shown for unsupported HPE and third-party transceivers installed in the switch and they are also identified with an asterisk in the output.

Command context

Manager (#)

Parameters

<INTERFACE-ID>

Specifies the name or range of an interface on the switch. Use the format `member/slot/port` (for example, `1/3/1`).

`detail`

Show detailed information for the interfaces.

`threshold-violations`

Show threshold violations for transceivers.

`[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

On the 6400 Switch Series, interface identification differs.

Showing summary transceiver information with identification of unsupported transceivers:

```
switch(config)# show interface transceiver
-------------------------------------------------------------------------
Port      Type           Product      Serial          Part
                         Number       Number          Number
-------------------------------------------------------------------------
1/1/1     SFP+SR         J9150A       MYxxxxxxxx      1990-3657
1/1/2     SFP+ER*        --           --              --
1/2/1     QSFP+SR4       JH233A       MYxxxxxxxx      2005-1234
1/2/2     QSFP+ER4*      --           --              --
1/3/1     SFP28DAC3      844477-B21   MYxxxxxxxx      77fc-7ce7

* unsupported transceiver
```

Showing detailed transceiver information:

```
switch(config)# show interface transceiver detail
Transceiver in 1/1/1
   Interface Name     : 1/1/1
   Type               : SFP+SR
   Connector Type     : LC
   Wavelength         : 850nm
   Transfer Distance  : 0m (SMF), 30m (OM1), 80m (OM2), 300m (OM3)
   Diagnostic Support : DOM
   Product Number     : J9150A
   Serial Number      : MYxxxxxxx
   Part Number        : 1990-3657

Status
   Temperature : 47.65C
   Voltage     : 3.31V
   Tx Bias     : 8.40mA
   Rx Power    : 0.08mW, -10.96dBm
   Tx Power    : 0.56mW, -2.49dBm

  Recent Alarms :
    Rx power low alarm
    Rx power low warning

  Recent Errors :
    Rx loss of signal

Transceiver in 1/1/2
   Interface Name     : 1/1/2
   Type               : unknown
   Connector Type     : ??
   Wavelength         : ??
   Transfer Distance  : ??
   Diagnostic Support : ??
   Product Number     : ??
   Serial Number      : ??
   Part Number        : ??

Transceiver in 1/2/1
   Interface Name     : 1/2/1
   Type               : QSFP+SR4
   Connector Type     : MPO
   Wavelength         : 850nm
   Transfer Distance  : 0m (SMF), 0m (OM1), 0m (OM2), 100m (OM3)
   Diagnostic Support : DOM
   Product Number     : JH233A
   Serial Number      : MYxxxxxxx
   Part Number        : 2005-1234

Status
   Temperature : 44.46C
   Voltage     : 3.30V

  ---------------------------------------------
            Tx Bias  Rx Power      Tx Power
  Channel#  (mA)     (mW/dBm)      (mW/dBm)
  ---------------------------------------------
  1         6.12     0.00, -inf    0.63, -1.95
  2         6.04     0.00, -inf    0.63, -2.00
  3         6.51     0.00, -inf    0.60, -2.16
  4         6.19     0.00, -inf    0.63, -1.94
```

```
  Recent Alarms :
    Channel 1 :
       Rx power low alarm
       Rx power low warning
    Channel 2 :
       Rx power low alarm
       Rx power low warning
    Channel 3 :
       Rx power low alarm
       Rx power low warning
    Channel 4 :
       Rx power low alarm
       Rx power low warning

  Recent Errors :
    Channel 1 :
       Rx Loss of Signal
    Channel 2 :
       Rx Loss of Signal
    Channel 3 :
       Rx Loss of Signal
    Channel 4 :
       Rx Loss of Signal

Transceiver in 1/2/2
   Interface Name     : 1/2/2
   Type               : unknown
   Connector Type     : ??
   Wavelength         : ??
   Transfer Distance  : ??
   Diagnostic Support : ??
   Product Number     : ??
   Serial Number      : ??
   Part Number        : ??

Transceiver in 1/3/1
   Interface Name     : 1/3/1
   Type               : SFP28DAC3
   Connector Type     : Copper Pigtail
   Transfer Distance  : 0.00km (SMF), 0m (OM1), 0m (OM2), 0m (OM3)
   Diagnostic Support : None
   Product Number     : 844477-B21
   Serial Number      : MYxxxxxxx
   Part Number        : 77fc-7ce7
```

Showing detailed transceiver information with identification of unsupported transceivers:

```
switch# show interface transceiver detail
Transceiver in 1/1/2
  Interface Name      : 1/1/2
  Type                : SFP+ER (unsupported)
  Connector Type      : LC
  Wavelength          : 3590nm
  Transfer Distance   : 80m (SMF), 0m (OM1), 0m (OM2), 0m (OM3)
  Diagnostic Support  : DOM
  Vendor Name         : INNOLIGHT
  Vendor Part Number  : TR-PX15Z-NHP
  Vendor Part Revision: 1A
  Vendor Serial number: MYxxxxxxx

Status
```

```
  Temperature : 28.88C
  Voltage     : 3.30V
  Tx Bias     : 65.53mA
  Rx Power    : 0.00mW, -inf
  Tx Power    : 1.47mW, 1.67dBm

 Recent Alarms:
  Rx Power low alarm
  Rx Power low warning

 Recent Errors:
  Rx loss of signal
```

Showing transceiver threshold-violations:

```
switch(config)# show interface transceiver threshold-violations
-------------------------------------------------------
Port      Type        Channel  Type(s) of Recent
                               Threshold Violation(s)
-------------------------------------------------------
1/1/1     SFP+SR               Tx bias high warning
                               50.52 mA > 40.00 mA
1/1/2     SFP+ER*              ??
1/2/1     QSFP+SR4    1        Tx power low alarm
                               -17.00 dBm < -0.50 dBm
                      2        Tx bias low warning
                               3.12 mA < 4.00 mA
1/2/2     QSFP+ER4*            ??
1/3/1     SFP28DAC3            n/a

* unsupported transceiver
```

show ip interface

Syntax

`show ip interface <INTERFACE-ID> [vsx-peer]`

Description

Shows status and configuration information for an IPv4 interface.

Command context

`Manager (#)`

Parameters

<INTERFACE-ID>

Specifies the name of an interface. Format: `member/slot/port`.

`[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

On the 6400 Switch Series, interface identification differs.

```
switch# show ip interface 1/1/1

Interface 1/1/1 is up
 Admin state is up
 Hardware: Ethernet, MAC Address: 70:72:cf:fd:e7:b4
 IPv4 address 192.168.1.1/24
 MTU 1500
 RX
          0 packets, 0 bytes
 TX
          0 packets, 0 bytes
```

show ip source-interface

Syntax

```
show ip source-interface {sflow |  tftp |  radius | tacacs | all} [vrf <VRF-NAME>]
     [vsx-peer]
```

Description

Shows single source IP address configuration settings.

Command context

Manager (#)

Parameters

`sflow | tftp | radius | tacacs | all`

Shows single source IP address configuration settings for a specific protocol. The `all` option shows the global setting that applies to all protocols that do not have an address set.

`vrf <VRF-NAME>`

Specifies the name of a VRF.

`[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

On the 6400 Switch Series, interface identification differs.

Showing single source IP address configuration settings for sFlow:

```
switch# show ip source-interface sflow

Source-interface Configuration Information
----------------------------------------
Protocol        Source Interface
```

```
--------        ----------------
 sflow            10.10.10.1
```

Showing single source IP address configuration settings for all protocols:

```
switch# show ip source-interface all

Source-interface Configuration Information
----------------------------------------
Protocol        Source Interface
--------        ----------------
 all             1/1/1
```

show ipv6 interface

Syntax

`show ipv6 interface <INTERFACE-ID> [vsx-peer]`

Description

Shows status and configuration information for an IPv6 interface.

Command context

Manager (#)

Parameters

<INTERFACE-ID>

Specifies an interface ID. Format: `member/slot/port`.

`[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

On the 6400 Switch Series, interface identification differs.

```
switch# show ipv6 interface 1/1/1

Interface 1/1/1 is up
 Admin state is up
 IPv6 address:
    2001:0db8:85a3:0000:0000:8a2e:0370:7334/24 [VALID]
 IPv6 link-local address: fe80::1e98:ecff:fee3:e800/64 (default)[VALID]
 IPv6 virtual address configured: none
 IPv6 multicast routing: disable
 IPv6 Forwarding feature: enabled
 IPv6 multicast groups locally joined:
    ff02::ff70:7334  ff02::ffe3:e800  ff02::1  ff02::1:ff00:0
```

```
   ff02::2
IPv6 multicast (S,G) entries joined: none
IPv6 MTU: 1524 (using link MTU)
IPv6 unicast reverse path forwarding: none
IPv6 load sharing: none
RX
         0 packets, 0 bytes
TX
         0 packets, 0 bytes
```

show ipv6 source-interface

Syntax

```
show ipv6 source-interface {sflow |  tftp |  radius | tacacs | all} [vrf <VRF-NAME>]
     [vsx-peer]
```

Description

Shows single source IP address configuration settings.

Command context

Manager (#)

Parameters

`sflow | tftp | radius | tacacs | all`

Shows single source IP address configuration settings for a specific protocol. The `all` option shows the global setting that applies to all protocols that do not have an address set.

`vrf <VRF-NAME>`

Specifies the name of a VRF.

`[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

On the 6400 Switch Series, interface identification differs.

Showing single source IP address configuration settings for sFlow:

```
switch# show ipv6 source-interface sflow

Source-interface Configuration Information
------------------------------------------
Protocol        Source Interface
--------        ----------------
 sflow            2001:DB8::1
```

Showing single source IP address configuration settings for all protocols:

```
switch# show ipv6 source-interface all

Source-interface Configuration Information
----------------------------------------
Protocol        Source Interface
--------        ----------------
 all             1/1/1
```

shutdown

Syntax

```
shutdown

no shutdown
```

Description

Disables an interface. Interfaces are disabled by default when created.

The `no` form of this command enables an interface.

Command context

`config-if`

Authority

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

Examples

Disabling an interface:

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

Enabling an interface:

```
switch(config-if)# no shutdown
```

The source IP address is determined by the system and is typically the IP address of the outgoing interface in the routing table. However, routing switches may have multiple routing interfaces and outgoing packets can potentially be sent by different paths at different times. This results in different source IP addresses.

AOS-CX provides a configuration model that allows the selection of an IP address to use as the source address for all outgoing traffic. This allows unique identification of the software application on the server site regardless of which local interface has been used to reach the destination server. The source interface selection supports selecting an IP address or interface name.

If the source interface and source IP are configured, Source IP will have priority.

Source-interface selection commands

Select a command from the list in the left navigation menu.

ip source-interface

Syntax

```
ip source-interface <PROTOCOL> <IP-ADDR> [vrf <VRF-NAME>]

no ip source-interface <PROTOCOL> <IP-ADDR> [vrf <VRF-NAME>]
```

Description

Configures the source-interface IPv4 address to use for the specified protocol. If a VRF is not given, the default VRF applies. If no interface option is given, the device floods through interfaces and VRFs to reach Aruba Central. Whichever reaches Aruba Central will be picked automatically.

The `no` form of this command removes all configurations.

Command context

`config`

Parameters

<PROTOCOL>

Specifies the protocol to configure.

`all`

Selects all protocols that can be configured by this command.

`central`

Selects Aruba Central.

`dhcp_relay`

Selects DHCP relay.

`dns`

Selects DNS.

`ntp`

Selects NTP.

`radius`

Selects radius.

`sflow`

Selects sFLow.

`simplivity`

Selects simplivity.

`syslog`

Selects syslog.

`tacacs`

Selects TACACS.

`tftp`

Selects TFTP.

<IP-ADDR>

Specifies the IPv4 address.

`vrf` *<VRF-NAME>*

Specifies the VRF name.

Authority

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

Examples

Configuring source-interface IPv4 10.1.1.1 to use for the TFTP protocol:

```
switch(config)# ip source-interface tftp 10.1.1.1
```

Configuring source-interface IPv4 10.1.1.2 to use for the TFTP protocol on VRF green :

```
switch(config)# ip source-interface tftp 10.1.1.2 vrf green
```

Removing source-interface IPv4 10.1.1.1 configuration for the TFTP protocol:

```
switch(config)# no ip source-interface tftp 10.1.1.1
```

Removing source-interface IPv4 10.1.1.2 configuration for TFTP protocol on VRF green:

```
switch(config)# no ip source-interface tftp 10.1.1.2 vrf green
```

Configuring source-interface IPv4 10.1.1.1 to use for the DNS protocol:

```
switch(config)# ip source-interface dns 10.1.1.1
```

Configuring source-interface IPv4 10.1.1.2 to use for the DNS protocl on VRF green :

```
switch(config)# ip source-interface dns 10.1.1.2 vrf green
```

Removing source-interface IPv4 10.1.1.1configuration for the DNS protocol:

```
switch(config)# no ip source-interface tftp 10.1.1.1
```

Removing source-interface IPv4 10.1.1.2 configuration for the DNS protocol on VRF green:

```
switch(config)# no ip source-interface dns 10.1.1.2 vrf green
```

ip source-interface interface

Syntax

```
ip source-interface <PROTOCOL> interface <IFNAME> [vrf <VRF-NAME>]

no ip source-interface <PROTOCOL> interface <IFNAME> [vrf <VRF-NAME>]
```

Description

Configures the IPv4 source-interface interface to use for the specified protocol. If a VRF is not given, the default VRF applies.

The `no` form of this command removes the specified configuration.

Command context

`config`

Parameters

<PROTOCOL>

Specifies the protocol to configure.

`all`
: Selects all protocols supported by this command.

`central`
: Selects Aruba Central.

`dhcp_relay`
: Selects DHCP relay.

`dns`
: Selects DNS.

`ntp`
: Selects NTP.

`radius`
: Selects radius.

`sflow`
: Selects sFLow.

`syslog`
: Selects syslog.

`tacacs`
: Selects TACACS.

`tftp`
: Selects TFTP.

`vrf` *<VRF-NAME>*

Specifies the VRF name.

<IFNAME>

Specifies the interface name.

Authority

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

Examples

Configuring IPv4 source-interface interface 1/1/1 to use for the TFTP protocol:

```
switch(config)# ip source-interface tftp interface 1/1/1
```

Configuring IPv4 source-interface interface 1/1/2 to use for the TFTP protocol on VRF green :

```
switch(config)# ip source-interface tftp interface 1/1/2 vrf green
```

Removing IPv4 source-interface 1/1/1configuration for the TFTP protocol:

```
switch(config)# no ip source-interface tftp interface 1/1/1
```

Removing source-interface interface 1/1/2 configuration for TFTP protocol on VRF green:

```
switch(config)# no ip source-interface tftp interface 1/1/2 vrf green
```

ipv6 source-interface

Syntax

`ipv6 source-interface <PROTOCOL> <IPV6-ADDR> [vrf <VRF-NAME>]`

`no source-interface <PROTOCOL> <IPV6-ADDR> [vrf <VRF-NAME>]`

Description

Configures the source-interface IPv6 address to use for the specified protocol. If a VRF is not given, the default VRF applies.

The `no` form of this command removes the specified protocol configuration.

Command context

`config`

Parameters

`<PROTOCOL>`

Specifies the protocol to configure.

`all`

Selects all protocols supported by this command.

`central`

Selects Aruba Central.

`ntp`

Selects NTP.

`radius`

Selects radius.

`sflow`

Selects sFLow.

`syslog`

Selects syslog.

`tacacs`

Selects TACACS.

`tftp`

Selects TFTP.

<IPV6-ADDR>

Specifies the IPv6 address.

`vrf` *<VRF-NAME>*

Specifies the VRF name.

Authority

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

Examples

Configuring source-interface IPv6 1111:2222 to use for the TFTP protocol:

```
switch(config)# ipv6 source-interface tftp 1111:2222
```

Configuring source-interface IPv6 1111:3333 to use for TFTP protocol on VRF green :

```
switch(config)# ipv6 source-interface tftp 1111:3333 vrf green
```

Removing source-interface IPv6 1111:2222 configuration for TFTP protocol:

```
switch(config)# no ipv6 source-interface tftp 1111:2222
```

Removing source-interface IPv6 1111:3333 configuration for TFTP protocol on VRF green:

```
switch(config)# no ipv6 source-interface tftp 1111:3333 vrf green
```

ipv6 source-interface interface

Syntax

`ipv6 source-interface` *<PROTOCOL>* `interface` *<IFNAME>* `[vrf` *<VRF-NAME>*`]`

`no ipv6 source-interface` *<PROTOCOL>* `interface` *<IFNAME>* `[vrf` *<VRF-NAME>*`]`

Description

Configures the IPv6 source-interface interface to use for the specified protocol. If a VRF is not given, the default VRF applies.

The `no` form of this command removes all configurations.

Command context

`config`

Parameters

<PROTOCOL>

Specifies the protocol to configure.

`all`

Selects all protocols supported by this command.

`central`

Selects Aruba Central.

`ntp`

Selects NTP.

`radius`

Selects radius.

`sflow`

Selects sFLow.

`syslog`

Selects syslog.

`tacacs`

Selects TACACS.

`tftp`

SelectsTFTP.

<IFNAME>

Specifies the interface name.

`vrf` *<VRF-NAME>*

Specifies the VRF name.

Authority

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

Examples

Configuring IPv6 source-interface interface 1/1/1 to use for the TFTP protocol :

```
switch(config)# ipv6 source-interface tftp interface 1/1/1
```

Configuring IPv6 source-interface interface 1/1/2 to use for the TFTP protocol on VRF green :

```
switch(config)# ipv6 source-interface tftp interface 1/1/2 vrf green
```

Removing IPv6 source-interface interface 1/1/1 configuration for the TFTP protocol:

```
switch(config)# no ipv6 source-interface tftp interface 1/1/1
```

Removing IPv6 source-interface interface 1/1/2 configuration for the TFTP protocol on VRF green:

```
switch(config)# no ipv6 source-interface tftp interface 1/1/2 vrf green
```

show ip source-interface

Syntax

`show ip source-interface` *<PROTOCOL>* [`vrf` *<VRF-NAME>* | *all-vrfs*]

Description

Displays the source interface information for all VRFs or a specific VRF.

If a VRF is not specified, the default is displayed.

Command context

Manager (#)

Parameters

<PROTOCOL>

Specifies the protocol to show.

`all`

Shows the source interface configuration for all other protocols.

`central`

Shows the source interface configuration for Aruba Central.

`dhcp relay`

Shows the source interface configuration for DHCP relay.

`dns`

Shows the source interface configuration for DNS.

`ntp`

Shows the source interface configuration for NTP.

`radius`

Shows the source interface configuration for radius.

`sflow`

Shows the source interface configuration for sFLow.

`syslog`

Shows the source interface configuration for syslog.

`tacacs`

Shows the source interface configuration for TACACS.

`tftp`

Shows the source interface configuration for TFTP.

`vrf` *<VRF-NAME>*

Specifies the VRF name.

all-vrfs

Shows the source interface configuration for all VRFs.

Authority

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

Examples

Displaying all source-interface protocol configurations for VRF red:

```
switch# show ip source-interface all vrf red
Source-interface Configuration Information
------------------------------------------------------------------
Protocol        Src-Interface       Src-IP                  VRF
------------------------------------------------------------------
all             1/1/1                                       red
switch#
```

Displaying all source-interface protocol configurations for default VRF:

```
switch# show ip source-interface all
Source-interface Configuration Information
------------------------------------------------------------------------
Protocol        Src-Interface       Src-IP                VRF
------------------------------------------------------------------------
all                                 1.1.1.1               default
switch#
```

Displaying all source-interface protocol configurations for all VRFs:

```
switch# show ip source-interface all all-vrfs
Source-interface Configuration Information
------------------------------------------------------------------------
Protocol        Src-Interface       Src-IP                VRF
------------------------------------------------------------------------
all                                 2.2.2.2               all-vrfs
all                                 1.1.1.1               default
all              1/1/1/1                                  red
switch#
```

show ipv6 source-interface

Syntax

`show ipv6 source-interface <PROTOCOL> [detail] [vrf <VRF-NAME> | all-vrfs]`

Description

Displays the IPV6 source interface information configured in the router for all VRFs or a specific VRF.

If a VRF is not specified, the default is displayed.

Command context

Manager (#)

Parameters

<PROTOCOL>

Specifies the protocol to show.

`all`

Shows the source interface configuration for all other protocols.

`central`

Shows the source interface configuration for Aruba Central.

`ntp`

Shows the source interface configuration for NTP.

`radius`

Shows the source interface configuration for radius.

`sflow`

Shows the source interface configuration for sFLow.

`syslog`

Shows the source interface configuration for syslog.

`tacacs`

Shows the source interface configuration for TACACS.

`tftp`

Shows the source interface configuration for TFTP.

`vrf <VRF-NAME>`

Specifies the VRF name.

all-vrfs

Shows the source interface configuration for all VRF.

Authority

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

Examples

Displaying all IPv6 source-interface protocol configurations for default VRF:

```
switch# show ipv6 source-interface all
Source-interface Configuration Information
------------------------------------------------------------------
Protocol        Src-Interface       Src-IP                VRF
------------------------------------------------------------------
all                                 1111:2222             default
switch#
```

Displaying all IPv6 source-interface protocol configuration for VRF red:

```
switch# show ipv6 source-interface all vrf red
Source-interface Configuration Information
------------------------------------------------------------------
Protocol        Src-Interface       Src-IP                VRF
------------------------------------------------------------------
all             1/1/1                                     red
switch#
```

Displaying all IPv6 source-interface protocol configurations for all VRFs:

```
switch# show ipv6 source-interface all all-vrfs
Source-interface Configuration Information
------------------------------------------------------------------
Protocol        Src-Interface       Src-IP                VRF
------------------------------------------------------------------
all                                 2.2.2.2:3.3.3.3       all-vrfs
all                                 1.1.1.1:2.2.2.2       default
all              1/1/1              2::2                  red
switch#
```

show running-config

Syntax

```
show running-config
```

Description

Displays the current running configuration.

Command context

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

Displaying the running configuration (only items of interest to source interface selection are shown in this example output command):

Aruba Central is the priority agent. If no command is specified for ip source-interface, Central will choose the command automatically if it is reachable on any of the known ports.

```
switch# show running-config
vrf green
ip source-interface tftp interface 1/1/2 vrf green
ip source-interface radius interface 1/1/2 vrf green
ip source-interface ntp interface 1/1/2 vrf green
ip source-interface tacacs interface 1/1/2 vrf green
ip source-interface dns interface 1/1/2 vrf green
ip source-interface central interface 1/1/2 vrf green
ip source-interface all interface 1/1/2 vrf green
ipv6 source-interface tftp 2222::3333 vrf green
ipv6 source-interface radius 2222::3333 vrf green
ipv6 source-interface ntp 2222::3333 vrf green
ipv6 source-interface tacacs 2222::3333 vrf green
ipv6 source-interface central 2222::3333 vrf green
ipv6 source-interface all 2222::3333 vrf green
ip source-interface tftp 10.20.3.1
ip source-interface radius 10.20.3.1
ip source-interface ntp 10.20.3.1
ip source-interface tacacs 10.20.3.1
ip source-interface dns 10.20.3.1
ip source-interface central 10.20.3.1
ip source-interface all 10.20.3.1
interface 1/1/1
     no shutdown
     ip address 10.20.3.1/24
interface 1/1/2
     vrf attach green
     ip address 20.1.1.1/24
     ipv6 address 2222::3333/64
interface 1/1/45
     no shutdown
     ip address 100.1.0.1/24
     ipv6 address 1111::2222/64
ip route 100.2.0.0/24 10.20.3.2
switch#
```

VLANs are primarily used to provide network segmentation at layer 2. VLANs enable the grouping of users by logical function instead of physical location. They make managing bandwidth usage within networks possible by:

- Allowing grouping of high-bandwidth users on low-traffic segments
- Organizing users from different LAN segments according to their need for common resources and individual protocols
- Improving traffic control at the edge of networks by separating traffic of different protocol types.
- Enhancing network security by creating subnets to control in-band access to specific network resources

VLANs are generally assigned on an organizational basis rather than on a physical basis. For example, a network administrator could assign all workstations and servers used by a particular workgroup to the same VLAN, regardless of their physical locations.

Hosts in the same VLAN can directly communicate with one another. A router or a Layer 3 switch is required for hosts in different VLANs to communicate with one another.

VLANs help reduce bandwidth waste, improve LAN security, and enable network administrators to address issues such as scalability and network management.

VLAN interfaces

Learn more about trunk and access interfaces, as well as the commands needed to implement typical VLAN configurations on different Aruba products.

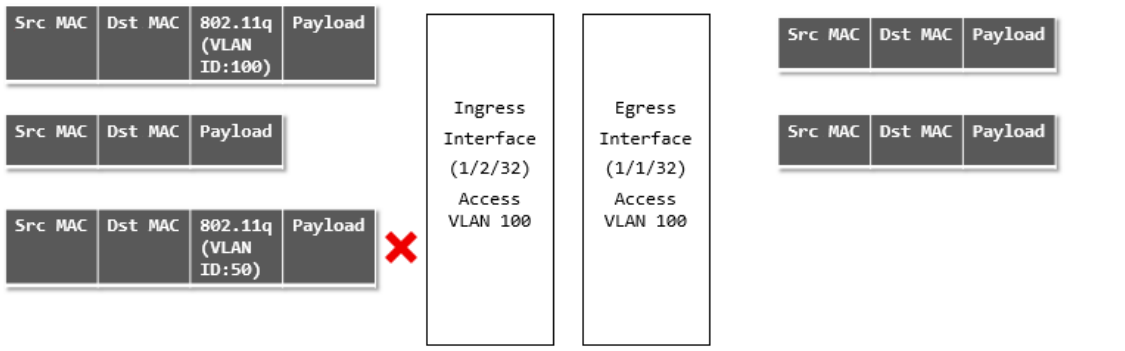
Access interface

An access interface carries traffic for a single VLAN ID. Access interfaces are generally used to connect end devices that do not support VLANs to the network. The devices connected to an access interface are not aware of the VLAN. Access interface can carry traffic on only one VLAN, either tagged or untagged.

Example

On the 6400 Switch Series, interface identification differs.

This example shows ingress and egress traffic behavior for an access interface.



;" />

- An ingress tagged frame with VLAN ID of 100 arrives on interface 1/2/32. The switch accepts this frame and sends it to its target address on interface 1/1/32, where it egresses untagged.
- An ingress untagged frame arrives on interface 1/2/32. The switch accepts this frame and sends it to its target address on interface 1/1/32, where it egresses untagged.
- An ingress tagged frame with VLAN ID of 50 arrives on interface 1/2/32. The switch drops this frame as VLAN ID 50 is not configured on the interface.

Trunk interface

A trunk interface can carry traffic for one or more VLAN IDs. In most cases, a trunk interface is used to transport data to other switches or routers.

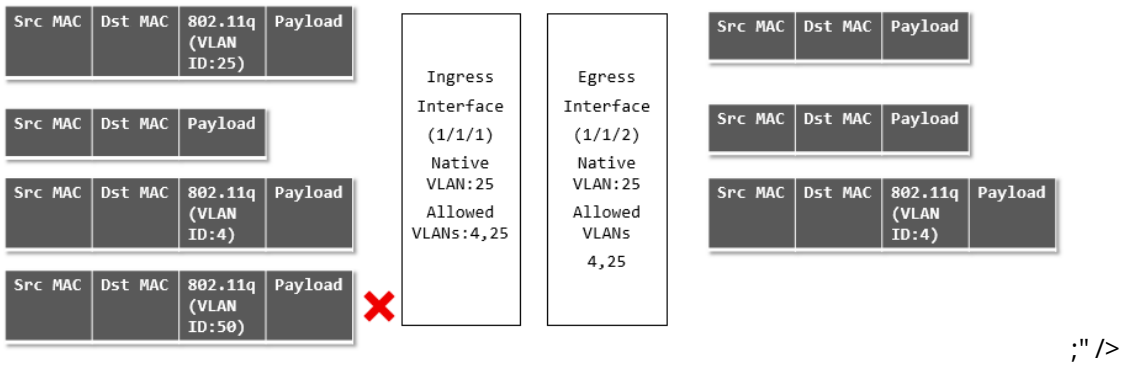
A trunk interface has two important settings:

- Native VLAN: This is the VLAN to which incoming untagged traffic is assigned. Only one VLAN can be assigned as the native VLAN. By default, VLAN 1 is assigned as the native VLAN for all trunk interfaces.
- Allowed VLANs: This is the list of VLANs that can be transported by the trunk. If the native VLAN is not included in the allowed list, all untagged frames that ingress on the trunk interface are dropped.

Example 1: Native untagged VLAN

On the 6400 Switch Series, interface identification differs.

This example shows ingress and egress traffic behavior when a trunk interface has a native untagged VLAN.



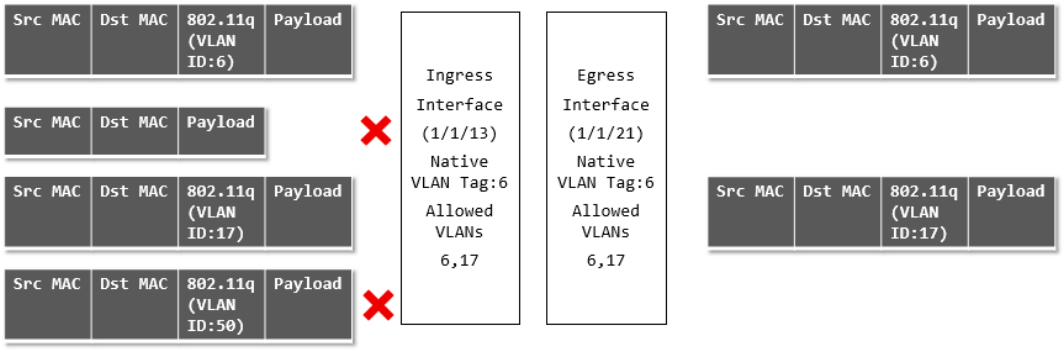
;" />

- An ingress tagged frame with VLAN ID of 25 arrives on interface 1/1/1. The switch accepts this frame and sends it to its target address on interface 1/1/2, where it egresses with a VLAN ID of 25 untagged since port 1/1/2 is configured with a native VLAN ID of 25.
- An ingress untagged frame arrives on interface 1/1/1. The switch accepts this frame and sends it to its target address on interface 1/1/2, where it egresses with a VLAN ID of 25 untagged since port 1/1/2 is configured with a native VLAN ID of 25.
- An ingress tagged frame with VLAN ID of 4 arrives on interface 1/1/1. The switch accepts this frame and sends it to its target address on interface 1/1/2, where it egresses with a VLAN ID of 4 tagged since port 1/1/2 is configured to allow traffic with a VLAN ID of 4.
- An ingress tagged frame with VLAN ID of 50 arrives on interface 1/1/1. The switch drops this frame as VLAN ID 50 is not in the allowed list for interface 1/1/1.

Example 2: Native tagged VLAN

On the 6400 Switch Series, interface identification differs.

This example shows ingress and egress traffic behavior when a trunk interface has a native tagged VLAN.



;" />

- An ingress tagged frame with VLAN ID of 6 arrives on interface 1/1/13. The switch accepts this frame and sends it to its target address on interface 1/1/21, where it egresses with a VLAN ID of 6 tagged since port 1/1/2 is configured with a native VLAN ID of 6.
- An ingress untagged frame arrives on interface 1/1/13. The switch drops this frame since the interface is configured as native tagged (all untagged frames a dropped in such a configuration).
- An ingress tagged frame with VLAN ID of 17 arrives on interface 1/1/13. The switch accepts this frame and sends it to its target address on interface 1/1/21, where it egresses with a VLAN ID of 17 tagged since port 1/1/2 is configured to allow traffic with a VLAN ID of 17.
- An ingress tagged frame with VLAN ID of 50 arrives on interface 1/1/13. The switch drops this frame as VLAN ID 50 is not in the allowed list for interface 1/1/13.

Traffic handling summary

VLAN configuration	Ingress traffic	Egress traffic
Access interface with: ■ VLAN ID = X	1. Untagged 2. Tagged with VLAN ID = X 3. Tagged with any other VLAN ID	1. Untagged on VLAN X 2. Untagged on VLAN X 3. Dropped
Trunk interface with: ■ Untagged Native VLAN ID = X ■ Allowed VLAN IDs = X, Y, Z	1. Untagged 2. Tagged with VLAN ID = X 3. Tagged with VLAN ID = Y 4. Tagged with VLAN ID = Z 5. Tagged with any other VLAN ID	1. Untagged on VLAN X 2. Untagged on VLAN X 3. Tagged on VLAN Y 4. Tagged on VLAN Z 5. Dropped
Trunk interface with: ■ Untagged Native VLAN ID = X ■ Allowed VLAN IDs = ALL	1. Untagged 2. Tagged with VLAN ID = X 3. Tagged with a VLAN ID defined on the switch 4. Tagged with a VLAN ID not defined on the switch	1. Untagged on VLAN X 2. Untagged on VLAN X 3. Tagged on the matching VLAN 4. Dropped
Trunk interface with: ■ Tagged Native VLAN ID = X ■ Allowed VLAN IDs = X, Y, Z	1. Untagged 2. Tagged with VLAN ID = X 3. Tagged with VLAN ID = Y	1. Dropped 2. Tagged on VLAN X 3. Tagged on VLAN Y

VLAN configuration	Ingress traffic	Egress traffic
	4. Tagged with VLAN ID = Z 5. Tagged with any other VLAN ID	4. Tagged on VLAN Z 5. Dropped
Trunk interface with: ■ Tagged Native VLAN ID = X ■ Allowed VLAN IDs = ALL	1. Untagged 2. Tagged with VLAN ID = X 3. Tagged with a VLAN ID defined on the switch 4. Tagged with a VLAN ID not defined on the switch	1. Dropped 2. Tagged on VLAN X 3. Tagged on the matching VLAN 4. Dropped
Trunk interface with: ■ Untagged Native VLAN ID = A ■ Allowed VLAN IDs = X, Y, Z	1. Untagged 2. Tagged with VLAN ID = X 3. Tagged with VLAN ID = Y 4. Tagged with VLAN ID = Z 5. Tagged with any other VLAN ID	1. Dropped 2. Tagged on VLAN X 3. Tagged on VLAN Y 4. Tagged on VLAN Z 5. Dropped

Comparing VLAN commands on PVOS, Comware, and AOS-CX

The following examples compare the commands needed to implement typical VLAN configurations on different HPE products.

On the 6400 Switch Series, interface identification differs.

AOS-CX	PVOS	Comware

`interface 1/1/1`
`vlan trunk native 1`
`vlan trunk allowed 10,30,50`
A native VLAN must be defined on the switch. By default, this is VLAN 1. Since only VLANs 10, 30, and 50 are allowed on the trunk, all untagged traffic is dropped.
	`interface A1`
	`  tagged vlan 10,30,50`
	`  no untagged vlan 1`
		`Interface G1/0/1`
		`  port link type trunk`
		`  port trunk permit vlan 10,30,50`
		`  port trunk pvid vlan 1`
		PVID 1 is the default setting.

AOS-CX	PVOS	Comware

`interface 1/1/1`
`vlan trunk native 10 tag`
`vlan trunk allowed 10,30,50`
Same as scenario 1, but allows untagged traffic on VLAN 10 as well.
	Not directly supported in PVOS. Scenario 1 is a workaround if there is no need to support untagged traffic.	Not directly supported in Comware. A possible workaround is:
		`interface g1/0/1`
		`  port link-mode bridge`
		`  port link-type hybrid`
		`  port hybrid protocol-vlan vlan 10`
		`  port hybrid vlan 10 tagged`
		`  port hybrid vlan 30 tagged`
		`  port hybrid vlan 50 tagged`

AOS-CX	PVOS	Comware

`interface 1/1/1`
`vlan trunk native 5`
`vlan trunk allowed 5, 10,30,50`
VLAN 5 must be allowed on the trunk so that untagged traffic is not dropped.
	`interface A1`
	`  untagged vlan 5`
	`  no tagged vlan 10,30,50`
		`interface G1/0/1`
		`  Port link-mode bridge`
		`  port link-type trunk`
		`  port trunk pvid vlan 5`
		`  port trunk permit vlan 5,10,30,50`
		`link-mode` is only needed on later Comware 7 devices. 5930 is port link-mode route by default. 5900 is bridge by default.

AOS-CX	PVOS	Comware

`interface 1/1/1`
`vlan access 5`
	`interface A1`
	`  untagged vlan 5`
		`interface G1/0/0`
		`  port link-mode bridge`
		`  port access vlan 5`

VLAN numbering

VLANs are numbered in the range 1 to 4094.

By default, VLAN 1 (the default VLAN) is associated with all interfaces on the switch. VLAN 1 cannot be removed from the switch.

Configuring VLANs

Learn procedures to create and enable a VLAN, assign a VLAN to an interface, and view VLAN configuration information.

Creating and enabling a VLAN

Procedure

1. Switch to the configuration context with the command `config`.
2. Create a new VLAN with the command `vlan`.

Example

This example creates **VLAN 10**. The VLAN is enabled by default.

```
switch# config
switch(config)# vlan 10
switch(config-vlan-10)#
```

Disabling a VLAN

Procedure

1. Switch to configuration context with the command `config`.
2. Switch to configuration context for the VLAN you want to disable with the command `vlan`.
3. Disable the VLAN with the command `shutdown`.

Example

This example disables **VLAN 10**.

```
switch(config)# config
switch(config)# vlan 10
switch(config-vlan-10)# shutdown
```

Assigning a VLAN to an interface

To use a VLAN, it must be assigned to an interface on the switch. VLANs can only be assigned to non-routed (layer 2) interfaces. All interfaces are routed (layer 3) by default when created. Use the `no routing` command to disable routing on an interface.

Assigning a VLAN ID to an access interface

Prerequisites

At least one defined VLAN.

Procedure

1. Switch to configuration context with the command `config`.
2. Switch to the interface that you want to define as an access interface with the command `interface`.
3. Configure the access interface and assign a VLAN ID with the command `vlan access`.

Examples

On the 6400 Switch Series, interface identification differs.

This example configures interface **1/1/2** as an access interface with VLAN ID set to **20**.

```
switch# config
switch(config)# vlan 20
switch(config-vlan-20)# exit
switch(config)# interface 1/1/2
switch(config-if)# vlan access 20
```

This example configures LAG **1** as an access interface with VLAN ID set to **30**.

```
switch# config
switch(config)# vlan 30
switch(config-vlan-30)# exit
switch(config)# interface lag 1
switch(config-lag-if)# no shutdown
switch(config-lag-if)# vlan access 30
```

Assigning a VLAN ID to a trunk interface

Prerequisites

At least one defined VLAN.

Procedure

1. Switch to configuration context with the command `config`.
2. Switch to the interface that you want to define as a trunk interface with the command `interface`.
3. Configure the trunk interface and assign a VLAN ID with the command `vlan trunk allowed`.

Examples

On the 6400 Switch Series, interface identification differs.

This example configures interface **1/1/2** as a trunk interface allowing traffic with VLAN ID set to **20**.

```
switch# config
switch(config)# vlan 20
switch(config-vlan-20)# exit
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk allowed 20
```

This example configures interface **1/1/2** as a trunk interface allowing traffic with VLAN IDs **2**, **3**, and **4**.

```
switch# config
switch(config)# vlan 2,3,4
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk allowed 2,3,4
```

This example configures interface **1/1/2** as a trunk interface allowing traffic with VLAN IDs **2** to **8**.

```
switch# config
switch(config)# vlan 2-8
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk allowed 2-8
```

This example configures interface **1/1/2** as a trunk interface allowing traffic with VLAN IDs **2** to **8** and **10**.

```
switch# config
switch(config)# vlan 2-8,10
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk allowed 2-8,10
```

This example configures interface **1/1/2** as a trunk interface allowing traffic on all configured VLAN IDs (20-100).

```
switch# config
switch(config)# vlan 20-100
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk allowed all
```

Assigning a native VLAN ID to a trunk interface

Prerequisites

At least one defined VLAN.

Procedure

1. Switch to configuration context with the command `config`.
2. Switch to the trunk interface to which you want to assign the native VLAN ID with the command `interface`.
3. Assign the native VLAN ID with the command `vlan trunk native`. If tagging is required, use the command `vlan trunk native tag`.
4. Allow traffic tagged with the native VLAN ID to be transported by the trunk using the command `vlan trunk allowed`.

Example

On the 6400 Switch Series, interface identification differs.

This example assigns native VLAN ID **20** to trunk interface **1/1/2**.

```
switch# config
switch(config)# vlan 20
switch(config-vlan-20)# exit
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk native 20
```

This example assigns native VLAN ID **40** to trunk interface **1/1/5**, enables tagging, and allows traffic with VLAN ID 40 to be transported by the trunk.

```
switch# config
switch(config)# vlan 40
switch(config-vlan-40)# exit
switch(config)# interface 1/1/5
switch(config-if)# vlan trunk native 40 tag
switch(config-if)# vlan trunk allow 40
```

Viewing VLAN configuration information

Prerequisites

At least one defined VLAN.

Procedure

1. View a summary of VLAN configuration information with the command `show vlan summary`.
2. View VLAN configuration settings with the command `show vlan`.
3. View VLANs configured for a specific layer 2 interface with the command `show vlan port`.
4. View the commands used to configure VLAN settings with the command `show running-config interface`.

Example

On the 6400 Switch Series, interface identification differs.

This example displays a summary of all VLANs.

```
switch# show vlan summary
Number of existing VLANs: 11
Number of static VLANs:   11
Number of dynamic VLANs:  0
```

This example displays configuration information for all defined VLANs.

```
switch# show vlan
-------------------------------------------------------------------------------
VLAN  Name             Status  Reason          Type      Interfaces
-------------------------------------------------------------------------------
1     DEFAULT_VLAN_1     up      ok              static    1/1/3-1/1/4
2     UserVLAN1          up      ok              static    1/1/1,1/1/3,1/1/5
3     UserVLAN2          up      ok              static    1/1/2-1/1/3,1/1/5-1/1/6
5     UserVLAN3          up      ok              static    1/1/3
10    TestNetwork        up      ok              static    1/1/3,1/1/5
11    VLAN11             up      ok              static    1/1/3
12    VLAN12             up      ok              static    1/1/3,1/1/6,lag1-lag2
13    VLAN13             up      ok              static    1/1/3,1/1/6
14    VLAN14             up      ok              static    1/1/3,1/1/6
20    ManagementVLAN     down    admin_down      static    1/1/3,1/1/10
```

This example displays configuration information for **VLAN 2**.

```
switch# show vlan 2
-------------------------------------------------------------------------------
VLAN  Name                     Status  Reason          Type      Interfaces
-------------------------------------------------------------------------------
2     UserVLAN1                up      ok              static    1/1/1,1/1/3,1/1/5
```

This example displays the VLANs configured on interface **1/1/3**.

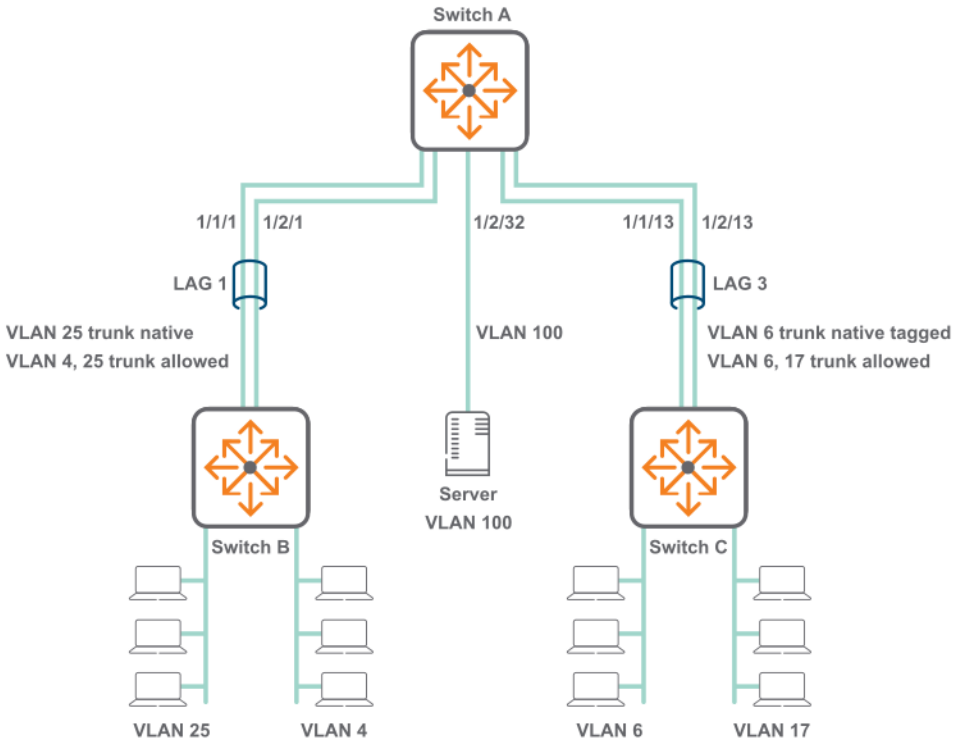
```
switch# show vlan port 1/1/3
---------------------------------------------------
VLAN  Name                            Mode
---------------------------------------------------
1     DEFAULT_VLAN_1                  native-untagged
2     UserVLAN1                       trunk
3     UserVLAN2                       trunk
5     UserVLAN3                       trunk
10    TestNetwork                     trunk
11    VLAN11                          trunk
12    VLAN12                          trunk
13    VLAN13                          trunk
14    VLAN14                          trunk
20    ManagementVLAN                  trunk
```

This example displays VLAN configuration commands for interface **1/1/16**.

```
switch#  show running-config interface 1/1/16
interface 1/1/16
   no routing
   vlan trunk native 108
   vlan trunk allowed all
   exit
```

VLAN scenario

This scenario shows how to assign VLAN IDs to access and trunk interfaces for the following deployment:



;" />

On the 6400 Switch Series, interface identification differs.

In this scenario, VLANs are used to isolate the traffic from different devices.

- VLAN 25 carries tagged and untagged traffic from computers connected to switch B.
- VLAN 4 carries tagged traffic from computers connected to switch B.
- VLAN 6 carries tagged and untagged traffic from computers connected to switch C.
- VLAN 17 carries tagged traffic from computers connected to switch C.
- VLAN 100 carries untagged traffic from the server.

Procedure

1. Execute the following commands on switch A and B.
   a. Create VLANs 4 and 25.

```
switch# config
switch(config)# vlan 4,25
```

   b. Define LAG **1** and assign the VLANs to it.

```
switch(config)# interface lag 1
switch(config-lag-if)# no shutdown
switch(config-lag-if)# vlan trunk native 25
switch(config-lag-if)# vlan trunk allowed 4,25
```

c. Add ports **1/1/1** and **1/2/1** to LAG **1**.

```
switch(config-lag-if)# interface 1/1/1
switch(config-if)# no shutdown
switch(config-if)# lag 1
switch(config-if)# interface 1/2/1
switch(config-if)# no shutdown
switch(config-if)# lag 1
```

2. Execute the following commands on switch A and C.
   a. Create VLANs 6 and 17.

```
switch# config
switch(config)# vlan 6,17
```

   b. Define LAG 3 and assign the VLANs to it.

```
switch(config)# interface lag 3
switch(config-lag-if)# no shutdown
switch(config-lag-if)# vlan trunk native 6 tag
switch(config-lag-if)# vlan trunk allowed 6,17
```

   c. Add ports **1/1/13** and **1/2/13** to LAG 3.

```
switch(config-lag-if)# interface 1/1/13
switch(config-if)# no shutdown
switch(config-if)# lag 3
switch(config-if)# interface 1/2/13
switch(config-if)# no shutdown
switch(config-if)# no routing
switch(config-if)# lag 3
```

3. Execute the following commands on switch A to configure the connection to the server.
   a. Configure interface **1/2/13** as an access interface with VLAN ID set to 100.

```
switch# config
switch (config)# vlan 100
switch(config-vlan-100)# interface 1/2/32
switch(config-if)# no shutdown
switch(config-if)# vlan access 100
switch(config-if)# exit
```

4. Verify VLAN configuration by running the command `show vlan`. For example:

```
switch# show vlan

--------------------------------------------------------------------------------
------------------------------
VLAN  Name                             Status  Reason               Type
Interfaces
```

```
--------------------------------------------------------------------------------
------------------------------
1     DEFAULT_VLAN_1                    down    no_member_port        default
4     VLAN4                             up      ok                    static
 lag1
6     VLAN6                             up      ok                    static
 lag3
17    VLAN17                            up      ok                    static
 lag3
25    VLAN25                            up      ok                    static
 lag1
100   VLAN100                           up      ok                    static
 1/2/32
```

5. Verify that the connection to the DHCP server is sending/receiving data with the command `show interface`. Check that the **Rx** and **Tx** fields are incrementing. For example:

```
switch# show interface 1/2/32
Interface 1/2/32 is up
 Admin state is up
 Description:
 Hardware: Ethernet, MAC Address: 70:72:cf:3a:8a:0b
 MTU 1500
 Type SFP+LR
 qos trust none
 Speed 10000 Mb/s
 Auto-Negotiation is off
 Input flow-control is off, output flow-control is off
 VLAN Mode: access
 Access VLAN: 100

 Rx
            20 input packets              1280 bytes
            0 input error                0 dropped
            0 CRC/FCS
 Tx
            9 output packets             1054 bytes
            0 input error                0 dropped
            0 collision
```

6. Verify LAG interface configuration with the command `show interface`. Check the fields admin state, MAC address, Aggregated-interfaces, VLAN Mode, Native VLAN, Allowed VLAN, Rx count, and Tx count. For example:

```
switch# show interface lag1
Aggregate-name lag1
 Description :
 Admin state           : up
 MAC Address           : 94:f1:28:21:63:00
 Aggregated-interfaces : 1/1/1 1/2/1
 Aggregation-key       : 1
 Speed                 : 1000 Mb/s
 L3 Counters: Rx Disabled, Tx Disabled
 qos trust none
 VLAN Mode: native-untagged
 Native VLAN: 25
```

```
 Allowed VLAN List: 4,25
 Rx
            10 input packets              1280 bytes
            0 input error                 0 dropped
            0 CRC/FCS
 Tx
            8 output packets              980 bytes
            0 input error                 0 dropped
            0 collision
```

```
switch# show interface lag3
Aggregate-name lag3
 Description :
 Admin state           : up
 MAC Address           : 94:f1:28:21:63:00
 Aggregated-interfaces : 1/1/13 1/2/13
 Aggregation-key       : 3
 Speed 1000 Mb/s
 L3 Counters: Rx Disabled, Tx Disabled
 qos trust none
 VLAN Mode: native-tagged
 Native VLAN: 6
 Allowed VLAN List: 6,17
 Rx
            19 input packets               1280 bytes
            0 input error                 0 dropped
            0 CRC/FCS
Tx
            15 output packets              1000 bytes
            0 input error                 0 dropped
0      Collision
```

7. Verify the physical interfaces (1/1/1, 1/2/1, 1/1/13, 1/2/13) with the command `show interface`. Check that the **Rx** and **Tx** fields are incrementing. For example:

```
switch# show interface 1/1/1
Interface 1/1/1 is up
 Admin state is up
 Description:
 Hardware: Ethernet, MAC Address: 94:f1:28:21:73:ff
 MTU 1500
 Type SFP+LR
 qos trust none
 Speed 1000 Mb/s
 Auto-Negotiation is off
 Input flow-control is off, output flow-control is off
 Rx
            6 input packets               620 bytes
            0 input error                 0 dropped
            0 CRC/FCS
 Tx
            4 output packets              422 bytes
            0 input error                 0 dropped
0      collision
```

VLAN commands

description

Syntax

`description <DESCRIPTION>`

Description

Specifies a descriptive for a VLAN.

Command context

`config-vlan-<VLAN-ID>`

Parameters

<DESCRIPTION>

Specifies a description for the VLAN.

Authority

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

Examples

Assigning a description to VLAN **20**:

```
switch(config)# vlan 20
switch(config-vlan-20)# description primary
```

name

Syntax

`name <VLAN-NAME>`

Description

Associates a name with a VLAN.

Command context

`config-vlan-<VLAN-ID>`

Parameters

<VLAN-NAME>

Specifies a name for a VLAN. Length: 1 to 32 alphanumeric characters, including underscore (_) and hyphen (-).

Authority

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

Examples

Assigning the name **backup** to VLAN **20**:

```
switch(config)# vlan 20
switch(config-vlan-20)# name backup
```

show capacities svi-count

Syntax

```
show capacities svi-count
```

Description

Shows the maximum number of SVIs supported by the switch.

Command context

Manager (#)

Authority

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

Examples

Showing switch SVI capacity:

```
switch# show capacities svi-count
System Capacities: Filter SVI count
Capacities Name                                                    Value
-------------------------------------------------------------------------
Maximum number of SVIs supported in the system                     128
```

show vlan

Syntax

`show vlan [<VLAN-ID>]` [vsx-peer]

Description

Displays configuration information for all VLANs or a specific VLAN.

Command context

`Manager (#)`

Parameters

<VLAN-ID>

Specifies a VLAN ID.

`[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

On the 6400 Switch Series, interface identification differs.

Displaying configuration information for VLAN **2**:

```
switch# show vlan 2
-------------------------------------------------------------------------------
VLAN  Name                 Status  Reason          Type      Interfaces
-------------------------------------------------------------------------------
2     UserVLAN1            up      ok              static    1/1/1,1/1/3,1/1/5
```

Displaying configuration information for all defined VLANs:

```
switch# show vlan
-------------------------------------------------------------------------------
VLAN  Name                 Status  Reason        Type      Interfaces
-------------------------------------------------------------------------------
1     DEFAULT_VLAN_1       up      ok            static    1/1/3-1/1/4
2     UserVLAN1            up      ok            static    1/1/1,1/1/3,1/1/5
3     UserVLAN2            up      ok            static    1/1/2-1/1/3,1/1/5-1/1/6
5     UserVLAN3            up      ok            static    1/1/3
10    TestNetwork          up      ok            static    1/1/3,1/1/5
11    VLAN11               up      ok            static    1/1/3
12    VLAN12               up      ok            static    1/1/3,1/1/6,lag1-lag2
13    VLAN13               up      ok            static    1/1/3,1/1/6
14    VLAN14               up      ok            static    1/1/3,1/1/6
20    ManagementVLAN       down    admin_down    static    1/1/3,1/1/10
```

show vlan port

Syntax

`show vlan port <INTERFACE-ID> [vsx-peer]`

Description

Displays the VLANs configured for a specific layer 2 interface.

Command context

Manager (`#`)

Parameters

`<INTERFACE-ID>`

Specifies an interface ID. Format: `member/slot/port`.

`[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

On the 6400 Switch Series, interface identification differs.

Displaying the VLANs configured on interface **1/1/3**:

```
switch# show vlan port 1/1/3
------------------------------------------------------
VLAN  Name                              Mode
------------------------------------------------------
1     DEFAULT_VLAN_1                    native-untagged
2     UserVLAN1                         trunk
3     UserVLAN2                         trunk
5     UserVLAN3                         trunk
10    TestNetwork                       trunk
11    VLAN11                            trunk
12    VLAN12                            trunk
13    VLAN13                            trunk
14    VLAN14                            trunk
20    ManagementVLAN                    trunk
```

show vlan summary

Syntax

```
show vlan summary [vsx-peer]
```

Description

Displays a summary of the VLAN configuration on the switch.

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

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

Displaying a summary of the VLAN configuration on the switch:

```
switch# show vlan summary
Number of existing VLANs: 11
Number of static VLANs:   11
Number of dynamic VLANs:  0
```

show vlan translation

Syntax

```
show vlan translation [interface <INTERFACE-NAME>] [vsx-peer]
```

Description

Shows a summary of all VLAN translations rules defined on the switch, or the rules defined for a specific interface.

Command context

Manager (#)

Parameters

`interface <INTERFACE-NAME>`

Specifies the name of a layer 2 interface. Format: `member/slot/port`.

`[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

On the 6400 Switch Series, interface identification differs.

Displaying a summary of all VLAN translations rules defined on the switch:

```
switch# show vlan translation
-------------------------------------------
Interface  VLAN-1      VLAN-2
-------------------------------------------
1/1/5      10          20
1/1/5      30          40
1/1/5      50          100
1/1/6      100         200

Total number of translation rules : 4
```

Displaying a summary of all VLAN translations rules defined on interface **1/1/5**:

```
switch# show vlan translation interface 1/1/5
-------------------------------------------
Interface  VLAN-1      VLAN-2
-------------------------------------------
1/1/5      10          20
1/1/5      30          40
1/1/5      50          100
```

shutdown

Syntax

`shutdown`

`no shutdown`

Description

Disables a VLAN. (By default, a VLAN is automatically enabled when it is created with the `vlan` command.)

The no form of this command enables a VLAN.

Command context

config-vlan-<*VLAN-ID*>

Authority

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

Examples

Enabling VLAN 20:

```
switch(config)# vlan 20
switch(config-vlan-20)# no shutdown
```

Disabling VLAN 20:

```
switch(config)# vlan 20
switch(config-vlan-20)# shutdown
```

system vlan-client-presence-detect

Supported on the 6300 and 6400 Switch Series only.

Syntax

```
system vlan-client-presence-detect

no system vlan-client-presence-detect
```

Description

Enables VNI mapped VLANs when detecting the presence of a client. When enabled, VNI mapped VLANs are *up* only if there are authenticated clients on the VLAN, or if the VLAN has statically configured ports and those ports are *up*. When not enabled, VNI mapped VLANs are always *up*.

The no form of this command disables detection of clients on VNI mapped VLANs.

Command context

config

Authority

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

Examples

Enabling detection of clients:

```
switch(config)# system vlan-client-presence-detect
```

Disabling detection of clients:

```
switch(config)# no system vlan-client-presence-detect
```

vlan

Syntax

```
vlan <VLAN-LIST>

no vlan <VLAN-LIST>
```

Description

Creates a VLAN and changes to the `config-vlan-id` context for the VLAN. By default, the VLAN is enabled. To disable a VLAN, use the `no shutdown` command.

If the specified VLAN exists, this command changes to the `config-vlan-id` context for the VLAN. If a range of VLANs is specified, the context does not change.

The `no` form of this command removes a VLAN. VLAN 1 is the default VLAN and cannot be deleted.

Command context

```
config
```

Parameters

<VLAN-LIST>

Specifies a single ID, or a series of IDs separated by commas (2, 3, 4), dashes (2-4), or both (2-4,6). Range: 1 to 4094.

Authority

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

Examples

Creating VLAN **20**:

```
switch(config)# vlan 20
switch(config-vlan-20)#
```

Removing VLAN **20**:

```
switch(config)# no vlan 20
```

Creating VLANs **2** to **8** and **10**:

```
switch(config)# vlan 2-8,10
```

Removing VLANs **2** to **8** and **10**:

```
switch(config)# no vlan 2-8,10
```

vlan access

Syntax

```
vlan access <VLAN-ID>

no vlan access [<VLAN-ID>]
```

Description

Creates an access interface and assigns an VLAN ID to it. Only one VLAN ID can be assigned to each access interface.

VLANs can only be assigned to a non-routed (layer 2) interface or LAG interface. By default, all interfaces are routed (layer 3) when created. Use the `no routing` command to disable routing on an interface and change the interface to a layer 2 interface.

The `no` form of this command removes an access VLAN from the interface in the current context and sets it to the default VLAN ID of 1.

Command context

`config-if`

Parameters

<VLAN-ID>

Specifies a single ID, or a series of IDs separated by commas (2, 3, 4), dashes (2-4), or both (2-4,6). Range: 1 to 4094.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Configuring interface **1/1/2** as an access interface with VLAN ID set to **20**:

```
switch(config)# interface 1/1/2
switch(config-if)# vlan access 20
```

Removing VLAN ID **20** from interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# no vlan access 20
```

or:

```
switch(config)# interface 1/1/2
switch(config-if)# no vlan access
```

vlan translate

Syntax

```
vlan translate <VLAN-1> <VLAN-2>
no vlan translate <VLAN-1> <VLAN-2>
```

Description

Defines a bidirectional VLAN translation rule that maps an external VLAN ID to an internal VLAN ID on a LAG or layer 2 interface. Applies to both incoming and outgoing traffic.

The `no` form of this command removes an existing VLAN translation rule on the current interface.

VLAN translation and MVRP cannot be enabled on the same interface.

Command context

```
config-if
config-lag-if
```

Parameters

<VLAN-1>

Specifies the number of an external VLAN. Range: 1 - 4000.

<VLAN-2>

Specifies the number of an internal VLAN. Range: 1 - 4000.

Authority

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

Examples

Translates external VLAN **200** to internal VLAN **20** on interface **1/1/2**.

```
switch# config
switch(config)# vlan 20
switch(config-vlan-20)# exit
switch(config)# interface 1/1/2
switch(config-if)# no routing
switch(config-if)# vlan trunk allowed 20
switch(config-if)# vlan translate 200 20
```

Translates external VLANs **100** and **300** to internal VLANs **10** and **20** on interface **1/1/2**.

```
switch# config
switch(config)# vlan 10,30
switch(config-vlan-20)# exit
switch(config)# interface 1/1/2
switch(config-if)# no routing
switch(config-if)# vlan trunk allowed 10,30
switch(config-if)# vlan translate 100 10
switch(config-if)# vlan translate 300 30
```

vlan trunk allowed

Syntax

```
vlan trunk allowed [<VLAN-LIST> | all]

no vlan trunk allowed [<VLAN-LIST>]
```

Description

Assigns a VLAN ID to an trunk interface. Multiple VLAN IDs can be assigned to a trunk interface. These VLAN IDs define which VLAN traffic is allowed across the trunk interface.

VLANs can be assigned only to a non-routed (layer 2) interface or LAG interface. By default, all interfaces are routed (layer 3) when created. Use the `no routing` command to disable routing on an interface.

The `no` form of this command removes one or more VLAN IDs from a trunk interface. When the last VLAN is removed from a trunk interface, the interface continues to operate in trunk mode, and will trunk all the VLANs currently defined on the switch, and any new VLANs defined in the future. To disable the trunk interface, use the command shutdown.

Command context

`config-if`

Parameters

<VLAN-LIST>

Specifies a single ID, or a series of IDs separated by commas (2, 3, 4), dashes (2-4), or both (2-4,6). Range: 1 to 4094.

`all`

Configures the trunk interface to allow all the VLANs currently configured on the switch and any new VLANs that are configured in the future.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Assigning VLANs **2**, **3**, and **4** to trunk interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk allowed 2,3,4
```

Assigning VLAN IDs **2** to **8** to trunk interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk allowed 2-8
```

Assigning VLAN IDs **2** to **8** and 10 to trunk interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk allowed 2-8,10
```

Removing VLAN IDs **2**, **3**, and **4** from trunk interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# no vlan trunk allowed 2,3,4
```

Removing all VLANs assigned to trunk interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# no vlan trunk allowed
```

vlan trunk native

Syntax

```
vlan trunk native <VLAN-ID>

no vlan trunk native [<VLAN-ID>]
```

Description

Assigns a native VLAN ID to a trunk interface. By default, VLAN ID 1 is assigned as the native VLAN ID for all trunk interfaces. VLANs can only be assigned to a non-routed (layer 2) interface or LAG interface. Only one VLAN ID can be assigned as the native VLAN.

When a native VLAN is defined, the switch automatically executes the `vlan trunk allowed all` command to ensure that the default VLAN is allowed on the trunk. To only allow specific VLANs on the trunk, issue the `vlan trunk allowed` command specifying only specific VLANs.

The `no` form of this command removes a native VLAN from a trunk interface and assigns VLAN ID 1 as its native VLAN.

Command context

`config-if`

Parameters

<VLAN-ID>

Specifies a VLAN ID. Range: 1 to 4094.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Assigning native VLAN ID **20** to trunk interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk native 20
```

Removing native VLAN **20** from trunk interface **1/1/2** and returning to the default VLAN 1 as the native VLAN.

```
switch(config)# interface 1/1/2
switch(config-if)# no vlan trunk native 20
```

or:

```
switch(config)# interface 1/1/2
switch(config-if)# no vlan trunk native
```

Assigning native VLAN ID **20** to trunk interface **1/1/2** and then removing it from the list of allowed VLANs. (Only allow VLAN 10 on the trunk.)

```
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk native 20
switch(config-if)# vlan trunk allowed 10
```

vlan trunk native tag

Syntax

```
vlan trunk native <VLAN-ID> tag

no vlan trunk native <VLAN-ID> tag
```

Description

Enables tagging on a native VLAN. Only incoming packets that are tagged with the matching VLAN ID are accepted. Incoming packets that are untagged are dropped except for BPDUs. Egress packets are tagged.

The `no` form of this command removes tagging on a native VLAN.

Command context

```
config-if
```

Parameters

<VLAN-ID>

Specifies the number of a VLAN. Range: 1 to 4094.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Enabling tagging on native VLAN **20** on trunk interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# vlan trunk native 20
switch(config-if)# vlan trunk native 20 tag
```

Removing tagging on native VLAN **20** assigned to trunk interface **1/1/2**:

```
switch(config)# interface 1/1/2
switch(config-if)# no vlan trunk native 20 tag
```

Enabling tagging on native VLAN **20** assigned to LAG trunk interface **2**:

```
switch(config)# interface lag 2
switch(config-lag-if)# vlan trunk native 20
switch(config-lag-if)# vlan trunk native 20 tag
```

voice

Syntax

```
voice

no voice
```

Description

Configures a VLAN as a voice VLAN.

The `no` form of this command removes voice configuration from a VLAN.

Command context

`config-vlan-<VLAN-ID>`

Authority

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

Examples

Configuring VLAN 10 as a voice VLAN:

```
switch(config)# vlan 10
switch(config-vlan-10)# voice
```

Removing voice from VLAN 10:

```
switch(config-vlan-10)# no voice
```

Checkpoints

A checkpoint is a snapshot of the running configuration of a switch and its relevant metadata during the time of creation. Checkpoints can be used to apply the switch configuration stored within a checkpoint whenever needed, such as to revert to a previous, clean configuration. Checkpoints can be applied to other switches of the same platform. A switch is able to store multiple checkpoints.

Checkpoint types

The switch supports two types of checkpoints:

- **System generated checkpoints**: The switch automatically generates a system checkpoint whenever a configuration change occurs.
- **User generated checkpoints**: The administrator can manually generate a checkpoint whenever required.

Maximum number of checkpoints

- Maximum checkpoints: 64 (including the startup configuration)
- Maximum user checkpoints: 32
- Maximum system checkpoints: 32

User generated checkpoints

User checkpoints can be created at any time, as long as one configuration difference exists since the last checkpoint was created. Checkpoints can be applied to either the running or startup configurations on the switch.

All user generated checkpoints include a time stamp to identify when a checkpoint was created.

A maximum of 32 user generated checkpoints can be created.

System generated checkpoints

System generated checkpoints are automatically created by default. Whenever a configuration change occurs, the switch starts a timeout counter (300 seconds by default). For each additional configuration change, the timeout counter is restarted. If the timeout expires with no additional configuration changes being made, the switch generates a new checkpoint.

System generated checkpoints are named with the prefix `CPC` followed by a time stamp in the format `<YYYYMMDDHHMMSS>`. For example: `CPC20170630073127`.

System checkpoints can be applied using the checkpoint rollback feature or copy command.

A maximum of 32 system checkpoints can be created. Beyond this limit, the newest system checkpoint replaces the oldest system checkpoint.

Supported remote file formats

You can restore a switch configuration by copying a switch configuration stored on a USB drive or a remote network device through SFTP/TFTP. The remote file formats that the switch supports depends on where you plan to restore the checkpoint.

Restoring a checkpoint to a...	File type supported
Running configuration	▪ CLI ▪ JSON ▪ Checkpoint
Startup configuration	▪ JSON ▪ Checkpoint
Specified checkpoint	Specified checkpoint

Rollback

The term rollback is used to refer to when a switch configuration is reverted to a pre-existing checkpoint.

For example, the following command applies the configuration from checkpoint `ckpt1`. All previous configurations are lost after the execution of this command: `checkpoint rollback ckpt1`

You can also specify the rollback of the running configuration or of the startup configuration with a specified checkpoint, as shown with the following command: `copy checkpoint <checkpoint-name> {running-config | startup-config}`

Checkpoint auto mode

Checkpoint auto mode configures the switch with failover support, causing it to automatically revert to a previous configuration if it becomes inoperable or inaccessible due to configuration changes that are being made.

After entering checkpoint auto mode, you have a set amount of time to add, remove, or modify the existing switch configuration. To save your changes, you must execute the `checkpoint auto confirm` command before the auto mode timer expires. If you do not execute the `checkpoint auto confirm` command within the specified time, all configuration changes you made are discarded and the running configuration reverts to the state it was before entering checkpoint auto mode.

Testing a switch configuration in checkpoint auto mode

Process overview:

1. Enable the checkpoint auto mode.
2. To save the configuration, enter the `checkpoint auto confirm` command before the specified time set in step 1.

Checkpoint commands

checkpoint auto

Syntax

`checkpoint auto <TIME-LAPSE-INTERVAL>`

Description

Starts auto checkpoint mode. In auto checkpoint mode, the switch temporarily saves the runtime configuration as a checkpoint only for the specified time lapse interval. Configuration changes must be saved before the interval expires, otherwise the runtime configuration is restored from the temporary checkpoint.

Command context

Manager (#)

Parameters

<TIME-LAPSE-INTERVAL>

Specifies the time lapse interval in minutes. Range: 1 to 60.

Authority

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

Usage

To save the runtime checkpoint permanently, run the `checkpoint auto confirm` command during the time lapse interval. The filename for the saved checkpoint is named `AUTO<YYYYMMDDHHMMSS>`. If the `checkpoint auto confirm` command is not entered during the specified time lapse interval, the previous runtime configuration is restored.

Examples

Confirming the auto checkpoint:

```
switch# checkpoint auto 20
Auto checkpoint mode expires in 20 minute(s)
switch# WARNING  Please "checkpoint auto confirm" within 2 minutes
switch# checkpoint auto confirm
checkpoint AUTO20170801011154 created
```

In this example, the runtime checkpoint was saved because the `checkpoint auto confirm` command was entered within the value set by the `time-lapse-interval` parameter, which was 20 minutes.

Not confirming the auto checkpoint:

```
switch# checkpoint auto 20
Auto checkpoint mode expires in 20 minute(s)
switch# WARNING  Please "checkpoint auto confirm" within 2 minutes
WARNING: Restoring configuration. Do NOT add any new configuration.
Restoration successful
```

In this example, the runtime checkpoint was reverted because the `checkpoint auto confirm` command was not entered within the value set by the `time-lapse-interval` parameter, which was 20 minutes.

checkpoint auto confirm

Syntax

```
checkpoint auto confirm
```

Description

Signals to the switch to save the running configuration used during the auto checkpoint mode. This command also ends the auto checkpoint mode.

Command context

Manager (#)

Authority

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

Usage

To save the runtime checkpoint permanently, run the `checkpoint auto confirm` command during the time lapse value set by the `checkpoint auto <TIME-LAPSE-INTERVAL>` command. The generated checkpoint name will be in the format `AUTO<YYYYMMDDHHMMSS>`. If the `checkpoint auto confirm` command is not entered during the specified time lapse interval, the previous runtime configuration is restored.

Examples

Confirming the auto checkpoint:

```
switch# checkpoint auto confirm
```

checkpoint diff

Syntax

```
checkpoint diff {<CHECKPOINT-NAME1> | running-config | startup-config}
     {<CHECKPOINT-NAME2> | running-config | startup-config}
```

Description

Shows the difference in configuration between two configurations. Compare checkpoints, the running configuration, or the startup configuration.

Command context

Manager (#)

Parameters

`{<CHECKPOINT-NAME1> | running-config | startup-config}`

Selects either a checkpoint, the running configuration, or the startup configuration as the baseline.

`{<CHECKPOINT-NAME2> | running-config | startup-config}`

Selects either a checkpoint, the running configuration, or the startup configuration to compare.

Authority

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

Usability

The output of the `checkpoint diff` command has several symbols:

- The plus sign (+) at the beginning of a line indicates that the line exists in the comparison but not in the baseline.
- The minus sign (-) at the beginning of a line indicates that the line exists in the baseline but not in the comparison.

Examples

In the following example, the configurations of checkpoints `cp1` and `cp2` are displayed before the `checkpoint diff` command, so that you can see the context of the `checkpoint diff` command.

```
switch# show checkpoint cp1
Checkpoint configuration:
!
!Version ArubaOS-CX XL.10.00.0002
!Schema version 0.1.8
module 1/1 product-number jl363a
!
!
!
!
!
!
!
vlan 1,200
interface 1/1/1
    no shutdown
    ip address 1.0.0.1/24
interface 1/1/2
    no shutdown
    ip address 2.0.0.1/24

switch# show checkpoint cp2
Checkpoint configuration:
!
!Version ArubaOS-CX XL.10.00.0002
!Schema version 0.1.8
module 1/1 product-number jl363a
!
!
!
!
!
!
!
vlan 1,200,300
interface 1/1/1
    no shutdown
    ip address 1.0.0.1/24
interface 1/1/2
    no shutdown
    ip address 2.0.0.1/24

switch# checkpoint diff cp1 cp2
--- /tmp/chkpt11501550258421    2017-08-01 01:17:38.420514016 +0000
+++ /tmp/chkpt21501550258421    2017-08-01 01:17:38.420514016 +0000
@@ -9,7 +9,7 @@
 !
 !
 !
-vlan 1,200
+vlan 1,200,300
 interface 1/1/1
     no shutdown
     ip address 1.0.0.1/24
```

checkpoint post-configuration

Syntax

```
checkpoint post-configuration

no checkpoint post-configuration
```

Description

Enables creation of system generated checkpoints when configuration changes occur. This feature is enabled by default.

The `no` form of this command disables system generated checkpoints.

Command context

Manager (`#`)

Authority

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

Usage

System generated checkpoints are automatically created by default. Whenever a configuration change occurs, the switch starts a timeout counter (300 seconds by default). For each additional configuration change, the timeout counter is restarted. If the timeout expires with no additional configuration changes being made, the switch generates a new checkpoint.

System generated checkpoints are named with the prefix `CPC` followed by a time stamp in the format `<YYYYMMDDHHMMSS>`. For example: `CPC20170630073127`.

System checkpoints can be applied using the checkpoint rollback feature or copy command.

A maximum of 32 system checkpoints can be created. Beyond this limit, the newest system checkpoint replaces the oldest system checkpoint.

Examples

Enabling system checkpoints:

```
switch(config)# checkpoint post-configuration
```

Disabling system checkpoints:

```
switch(config)# no checkpoint post-configuration
```

checkpoint post-configuration timeout

Syntax

```
checkpoint post-configuration timeout <TIMEOUT>

no checkpoint post-configuration timeout <TIMEOUT>
```

Description

Sets the timeout for the creation of system checkpoints. The timeout specifies the amount of time since the latest configuration for the switch to create a system checkpoint.

The `no` form of this command resets the timeout to 300 seconds, regardless of the value of the *`<TIMEOUT>`* parameter.

Command context

Manager (`#`)

Parameters

`timeout` *`<TIMEOUT>`*

Specifies the timeout in seconds. Range: 5 to 600. Default: 300.

Authority

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

Examples

Setting the timeout for system checkpoints to 60 seconds:

```
switch(config)# checkpoint post-configuration timeout 60
```

Resetting the timeout for system checkpoints to 300 seconds:

```
switch(config)# no checkpoint post-configuration timeout 1
```

checkpoint rename

Syntax

```
checkpoint rename <OLD-CHECKPOINT-NAME> <NEW-CHECKPOINT-NAME>
```

Description

Renames an existing checkpoint.

Command context

Manager (#)

Parameters

<OLD-CHECKPOINT-NAME>

Specifies the name of an existing checkpoint to be renamed.

<NEW-CHECKPOINT-NAME>

Specifies the new name for the checkpoint. The checkpoint name can be alphanumeric. It can also contain underscores (_) and dashes (-).

Do not start the checkpoint name with `CPC` because it is used for system-generated checkpoints.

Authority

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

Examples

Renaming checkpoint **ckpt1** to **cfg001**:

```
switch# checkpoint rename ckpt1 cfg001
```

checkpoint rollback

Syntax

```
checkpoint rollback {<CHECKPOINT-NAME> | startup-config}
```

Description

Applies the configuration from a pre-existing checkpoint or the startup configuration to the running configuration.

Command context

Manager (#)

Parameters

<CHECKPOINT-NAME>

Specifies a checkpoint name.

`startup-config`

Specifies the startup configuration.

Authority

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

Examples

Applying a checkpoint named `ckpt1` to the running configuration:

```
switch# checkpoint rollback ckpt1
Success
```

Applying a startup checkpoint to the running configuration:

```
switch# checkpoint rollback startup-config
Success
```

copy checkpoint *<CHECKPOINT-NAME>* *<REMOTE-URL>*

Syntax

`copy checkpoint` *<CHECKPOINT-NAME>* *<REMOTE-URL>* `[vrf` *<VRF-NAME>*`]`

Description

Copies a checkpoint configuration to a remote location as a file. The configuration is exported in checkpoint format, which includes switch configuration and relevant metadata.

Command context

Manager (#)

Parameters

<CHECKPOINT-NAME>

Specifies the name of a checkpoint.

<REMOTE-URL>

Specifies the remote destination and filename using the syntax: `{tftp | sftp}://`*<IP-ADDRESS>* `[:`*<PORT-NUMBER>*`][;blocksize=`*<BLOCKSIZE-VALUE>*`]/`*<FILE-NAME>*

`vrf` *<VRF-NAME>*

Specifies a VRF name.

Authority

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

Examples

Copying checkpoint configuration to remote file through TFTP:

```
switch# copy checkpoint ckpt1 tftp://192.168.1.10/ckptmeta vrf default
##################################################################################### 100.0%
Success
```

Copying checkpoint configuration to remote file through SFTP:

```
switch# copy checkpoint ckpt1 sftp://root@192.168.1.10/ckptmeta vrf default
The authenticity of host '192.168.1.10 (192.168.1.10)' can't be established.
ECDSA key fingerprint is SHA256:FtOm6Uxuxumil7VCwLnhz92H9LkjY+eURbdddOETy50.
Are you sure you want to continue connecting (yes/no)? yes
root@192.168.1.10's password:
sftp> put /tmp/ckptmeta ckptmeta
Uploading /tmp/ckptmeta to /root/ckptmeta
Warning: Permanently added '192.168.1.10' (ECDSA) to the list of known hosts.
Connected to 192.168.1.10.
Success
```

copy checkpoint <*CHECKPOINT-NAME*> {running-config | startup-config}

Syntax

```
copy checkpoint <CHECKPOINT-NAME> {running-config | startup-config}
```

Description

Copies an existing checkpoint configuration to the running configuration or to the startup configuration.

Command context

Manager (#)

Parameters

<CHECKPOINT-NAME>

Specifies the name of an existing checkpoint.

`{running-config | startup-config}`

Selects whether the running configuration or the startup configuration receives the copied checkpoint configuration. If the startup configuration is already present, the command overwrites the startup configuration.

Authority

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

Examples

Copying **ckpt1** checkpoint to the running configuration:

```
switch# copy checkpoint ckpt1 running-config
Success
```

Copying **ckpt1** checkpoint to the startup configuration:

```
switch# copy checkpoint ckpt1 startup-config
Success
```

copy checkpoint <*CHECKPOINT-NAME*> <*STORAGE-URL*>

Syntax

`copy checkpoint <CHECKPOINT-NAME> <STORAGE-URL>`

Description

Copies an existing checkpoint configuration to a USB drive. The file format is defined when the checkpoint was created.

Command context

Manager (`#`)

Parameters

<CHECKPOINT-NAME>

Specifies the name of the checkpoint to copy. The checkpoint name can be alphanumeric. It can also contain underscores (_) and dashes (-).

<STORAGE-URL>>

Specifies the name of the target file on the USB drive using the following syntax: `usb:/<FILE>`

The USB drive must be formatted with the FAT file system.

Authority

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

Examples

Copying the `test` checkpoint to the `testCheck` file on the USB drive:

```
switch# copy checkpoint test usb:/testCheck
Success
```

copy <*REMOTE-URL*> checkpoint <*CHECKPOINT-NAME*>

Syntax

`copy <REMOTE-URL> checkpoint <CHECKPOINT-NAME> [vrf <VRF-NAME>]`

Description

Copies a remote configuration file to a checkpoint. The remote configuration file must be in checkpoint format.

Command context

Manager (`#`)

Parameters

<REMOTE-URL>

Specifies a remote file using the following syntax: `{tftp | sftp}://<IP-ADDRESS>[:<PORT-NUMBER>][;blocksize=<BLOCKSIZE-VALUE>]/<FILE-NAME>>`

<CHECKPOINT-NAME>

Specifies the name of the target checkpoint. The checkpoint name can be alphanumeric. It can also contain underscores (_) and dashes (-). Required.

Do not start the checkpoint name with `CPC` because it is used for system-generated checkpoints.

`vrf <VRF-NAME>`

Specifies a VRF name. Default: `default`.

Authority

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

Examples

Copying a checkpoint format file to checkpoint **ckpt5** on the default VRF:

```
switch# copy tftp://192.168.1.10/ckptmeta checkpoint ckpt5
##############################################################################  100.0%
100.0%
Success
```

copy <*REMOTE-URL*> {running-config | startup-config}

Syntax

`copy <REMOTE-URL> {running-config | startup-config } [vrf <VRF-NAME>]`

Description

Copies a remote file containing a switch configuration to the running configuration or to the startup configuration.

Command context

Manager (`#`)

Parameters

`<REMOTE-URL>`

Specifies a remote file with the following syntax: `{tftp | sftp}://<IP-ADDRESS>[:<PORT-NUMBER>] [;blocksize=<BLOCKSIZE-VALUE>]/<FILE-NAME>`

`{running-config | startup-config}`

Selects whether the running configuration or the startup configuration receives the copied checkpoint configuration. If the startup configuration is already present, the command overwrites the startup configuration.

`vrf <VRF-NAME>`

Specifies the name of a VRF. Default: default.

Authority

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

Usage

The switch copies only certain file types. The format of the file is automatically detected from contents of the file. The `startup-config` option only supports the JSON file format and checkpoints, but the `running-config` option supports the JSON and CLI file formats and checkpoints.

When a file of the CLI format is copied, it overwrites the running configuration. The CLI command does not clear the running configuration before applying the CLI commands. All of the CLI commands in the file are applied line-by-line. If a particular CLI command fails, the switch logs the failure and it continues to the next line in the CLI configuration. The event log (`show events -d hpe-config`) provides information as to which command failed.

Examples

Copying a JSON format file to the running configuration:

```
switch# copy tftp://192.168.1.10/runjson running-config
################################################################################ 100.0%
Configuration may take several minutes to complete according to configuration file
size
 --0%----10%----20%----30%----40%----50%----60%----70%----80%----90%----100%--
Success
```

Copying a CLI format file to the running configuration with an error in the file:

```
switch# copy tftp://192.168.1.10/runcli running-config
################################################################################ 100.0%
Configuration may take several minutes to complete according to configuration file
size
 --0%----10%----20%----30%----40%----50%----60%----70%----80%----90%----100%--
Some of the configuration lines from the file were NOT applied. Use 'show
events -d hpe-config' for more info.
```

Copying a CLI format file to the startup configuration:

```
switch# copy tftp://192.168.1.10/startjson startup-config
################################################################################ 100.0%
100.0%
Success
```

Copying an unsupported file format to the startup configuration:

```
switch# copy tftp://192.168.1.10/startfile startup-config
################################################################################ 100.0%
100.0%
unsupported file format
```

copy running-config {startup-config | checkpoint <*CHECKPOINT-NAME*>}

Syntax

`copy running-config {startup-config | checkpoint <CHECKPOINT-NAME>}`

Description

Copies the running configuration to the startup configuration or to a new checkpoint. If the startup configuration is already present, the command overwrites the existing startup configuration.

Command context

Manager (#)

Parameters

`startup-config`

Specifies that the startup configuration receives a copy of the running configuration.

`checkpoint <CHECKPOINT-NAME>`

Specifies the name of a new checkpoint to receive a copy of the running configuration. The checkpoint name can be alphanumeric. It can also contain underscores (_) and dashes (-).

Do not start the checkpoint name with `CPC` because it is used for system-generated checkpoints.

Authority

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

Examples

Copying the running configuration to the startup configuration:

```
switch# copy running-config startup-config
Success
```

Copying the running configuration to a new checkpoint named **ckpt1**:

```
switch# copy running-config checkpoint ckpt1
Success
```

copy {running-config | startup-config} *<REMOTE-URL>*

Syntax

`copy {running-config | startup-config} <REMOTE-URL> {cli | json} [vrf <VRF-NAME>]`

Description

Copies the running configuration or the startup configuration to a remote file in either CLI or JSON format.

Command context

Manager (`#`)

Parameters

`{running-config | startup-config}`

Selects whether the running configuration or the startup configuration is copied to a remote file.

`<REMOTE-URL>`

Specifies the remote file using the syntax: `{tftp | sftp}://<IP-ADDRESS>[:<PORT-NUMBER>][;blocksize=<BLOCKSIZE-VALUE>]/<FILE-NAME>`

`{cli | json}`

Selects the remote file format: P: CLI or JSON.

`vrf <VRF-NAME>`

Specifies the name of a VRF. Default: default.

Authority

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

Examples

Copying a running configuration to a remote file in CLI format:

```
switch# copy running-config tftp://192.168.1.10/runcli cli
################################################################################ 100.0%
Success
```

Copying a running configuration to a remote file in JSON format:

```
switch# copy running-config tftp://192.168.1.10/runjson json
################################################################################ 100.0%
Success
```

Copying a startup configuration to a remote file in CLI format:

```
switch# copy startup-config sftp://root@192.168.1.10/startcli cli
root@192.168.1.10's password:
sftp> put /tmp/startcli startcli
Uploading /tmp/startcli to /root/startcli
Connected to 192.168.1.10.
Success
```

Copying a startup configuration to a remote file in JSON format:

```
switch# copy startup-config sftp://root@192.168.1.10/startjson json
root@192.168.1.10's password:
sftp> put /tmp/startjson startjson
Uploading /tmp/startjson to /root/startjson
Connected to 192.168.1.10.
Success
```

copy {running-config | startup-config} *<STORAGE-URL>*

Syntax

`copy {running-config | startup-config} <STORAGE-URL> {cli | json}`

Description

Copies the running configuration or a startup configuration to a USB drive.

Command context

Manager (#)

Parameters

`{running-config | startup-config}`

Selects the running configuration or the startup configuration to be copied to the switch USB drive.

<STORAGE-URL>

Specifies a remote file with the following syntax: `usb:/<file>`

`{cli | json}`

Selects the format of the remote file: CLI or JSON.

Authority

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

Usage

The switch supports JSON and CLI file formats when copying the running or starting configuration to the USB drive. The USB drive must be formatted with the FAT file system.

The USB drive must be enabled and mounted with the following commands:

```
switch(config)# usb
switch(config)# end
switch# usb mount
```

Examples

Copying a running configuration to a file named `runCLI` on the USB drive:

```
switch# copy running-config usb:/runCLI cli
Success
```

Copying a startup configuration to a file named `startCLI` on the USB drive:

```
switch# copy startup-config usb:/startCLI cli
Success
```

copy startup-config running-config

Syntax

```
copy startup-config running-config
```

Description

Copies the startup configuration to the running configuration.

Command context

Manager (#)

Authority

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

Examples

```
switch# copy startup-config running-config
Success
```

copy <*STORAGE-URL*> running-config

Syntax

```
copy <STORAGE-URL> {running-config | startup-config | checkpoint <CHECKPOINT-NAME>}
```

Description

This command copies a specified configuration from the USB drive to the running configuration, to a startup configuration, or to a checkpoint.

Command context

Manager (#)

Parameters

<STORAGE-URL>

Specifies the name of a configuration file on the USB drive with the syntax: `usb:/<FILE>`

`running-config`

Specifies that the configuration file is copied to the running configuration. The file must be in CLI, JSON, or checkpoint format or the copy will fail. the copy will not work.

`startup-config`

Specifies that the configuration file is copied to the startup configuration. The switch stores this configuration between reboots. The startup configuration is used as the operating configuration following a reboot of the switch. The file must be in JSON or checkpoint format or the copy will fail.

`checkpoint <CHECKPOINT-NAME>`

Specifies the name of a new checkpoint file to receive a copy of the configuration. The configuration file on the USB drive must be in checkpoint format.

Do not start the checkpoint name with `CPC` because it is used for system-generated checkpoints.

Authority

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

Usage

This command requires that the USB drive is formatted with the FAT file system and that the file be in the appropriate format as follows:

- `running-config`: This option requires the file on the USB drive be in CLI, JSON, or checkpoint format.
- `startup-config`: This option requires the file on the USB drive be in JSON or checkpoint format.
- `checkpoint <checkpoint-name>`: This option requires the file on the USB drive be in checkpoint format.

Examples

Copying the file **runCli** from the USB drive to the running configuration:

```
switch# copy usb:/runCli running-config
Configuration may take several minutes to complete according to configuration
file size
--0%----10%----20%----30%----40%----50%----60%----70%----80%----90%----100%--
Success
```

Copying the file **startUp** from the USB drive to the startup configuration:

```
switch# copy usb:/startUp startup-config
Success
```

Copying the file **testCheck** from the USB drive to the **abc** checkpoint:

```
switch# copy usb:/testCheck checkpoint abc
Success
```

erase {checkpoint <*CHECKPOINT-NAME*> | startup-config | all}

Syntax

```
erase {checkpoint <CHECKPOINT-NAME> | startup-config | all}
```

Description

Deletes an existing checkpoint, startup configuration, or all checkpoints.

Command context

Manager (#)

Parameters

`checkpoint <CHECKPOINT-NAME>`

Specifies the name of a checkpoint.

`startup-config`

Specifies the startup configuration.

`all`

Specifies all checkpoints.

Authority

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

Examples

Erasing checkpoint **ckpt1**:

```
switch# erase checkpoint ckpt1
```

Erasing the startup configuration:

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

Erasing all checkpoints:

```
switch# erase checkpoint all
```

show checkpoint <*CHECKPOINT-NAME*>

Syntax

```
show checkpoint <CHECKPOINT-NAME> [json]
```

Description

Shows the configuration of a checkpoint.

Command context

Manager (#)

Parameters

`checkpoint <CHECKPOINT-NAME>`

Specifies the name of a checkpoint.

`[json]`

Specifies that the output is displayed in JSON format.

Authority

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

Examples

Showing the configuration of the `ckpt1` checkpoint in non-JSON format:

```
switch# show checkpoint ckpt1
Checkpoint configuration:
!
!Version ArubaOS-CX PL.10.07.0000K-75-g55e5193
!export-password: default
lacp system-priority 65535
user admin group administrators password ciphertext
AQBapQjwipebv36io0jFfde7ZzrHckncal1D+3n8XFTZKQdmYgAAADEtYOeHSme93xzdD0uz6Vr9Kl+XBzB+
2GB0UBxSF7rvgN2x8KSgkqv7iqXVQ0Te6LkSMnH4BdNaT3Bf25qyvOqmr4YakO1V3rg8zAOADkPktQD8joTH
XflzwomoIzcmv/uX
cli-session
    timeout 0
!
!
!
!
ssh server vrf default
vlan 1
spanning-tree
interface lag 1
    no shutdown
    vlan access 1
interface lag 128
    no shutdown
    vlan access 1
interface lag 129
    shutdown
    vlan access 1
    lacp mode active
interface 1/1/1
    no shutdown
    lag 128
    lacp port-id 65535
interface 1/1/2
    no shutdown
    vlan access 1
interface 1/1/3
    no shutdown
    vlan access 1
interface 1/1/4
    no shutdown
    vlan access 1
interface 1/1/5
    no shutdown
    vlan access 1
interface 1/1/6
    no shutdown
    vlan access 1
interface 1/1/7
    no shutdown
```

```
    vlan access 1
interface 1/1/8
    no shutdown
    vlan access 1
interface 1/1/9
    no shutdown
    vlan access 1
interface 1/1/10
    no shutdown
    vlan access 1
interface 1/1/11
    no shutdown
    vlan access 1
interface 1/1/12
    no shutdown
    vlan access 1
interface 1/1/13
    no shutdown
    vlan access 1
interface 1/1/14
    no shutdown
    vlan access 1
interface 1/1/15
    no shutdown
    vlan access 1
interface 1/1/16
    no shutdown
    vlan access 1
interface vlan 1
    ip dhcp
snmp-server vrf default
!
!
!
!
!
https-server vrf default
```

Showing the configuration of the `ckpt1` checkpoint in JSON format:

```
switch# show checkpoint ckpt1 json
Checkpoint configuration:
{
    "AAA_Server_Group": {
        "local": {
            "group_name": "local"
        },
        "none": {
            "group_name": "none"
        }
    },
...
...
...
...
```

show checkpoint post-configuration

Syntax

```
show checkpoint post-configuration
```

Description

Shows the configuration settings for creating system checkpoints.

Command context

Manager (#)

Authority

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

Examples

```
switch# show checkpoint post-configuration

Checkpoint Post-Configuration feature
-------------------------------------

 Status                : enabled
 Timeout (sec) : 300
```

show checkpoint list

Syntax

`show checkpoint list {all | <START-DATE> <END-DATE>}`

Description

Shows a list of saved checkpoints.

Command context

Manager (#)

Parameters

`all`

Shows a detailed list of all saved checkpoints.

<START-DATE>

Specifies the starting date for the range of saved checkpoints to show. Format: `YYYY-MM-DD`.

<END-DATE>

Specifies the ending date for the range of saved checkpoints to show. Format: `YYYY-MM-DD`.

Authority

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

Examples

Showing the list of saved checkpoints:

```
switch# show checkpoint list
ckpt1
ckpt2
ckpt3
startup-config
AUTO20170308214100
```

Showing a detailed list of all saved system and user checkpoints:

```
switch# show checkpoint list all
|NAME                  |TYPE          |WRITER   |DATE(UTC)              |HARDWARE
|IMAGE VERSION       |
|CPC20171017195548     |checkpoint    |System   |2020-10-17T19:55:48Z   |00000
|xx.10.0x.xxxxX-1-g7691be0|
|ckpt23                |checkpoint    |User     |2020-10-17T20:09:00Z   |00000
|xx.10.0x.xxxxX-1-g7691be0|
|ckpt24                |checkpoint    |User     |2020-10-17T20:09:09Z   |00000
|xx.10.0x.xxxxX-1-g7691be0|
|ckpt30                |checkpoint    |User     |2020-10-17T20:10:36Z   |00000
|xx.10.0x.xxxxX-1-g7691be0|
|startup-config-backup|checkpoint     |System   |2020-10-17T20:17:56Z   |00000
|xx.10.0x.xxxxX-1-g7691be0|
|CPC20171017201712     |checkpoint    |System   |2020-10-17T20:19:12Z   |00000
|xx.10.0x.xxxxAZ-10-g4c6b4446bd6|
|ckpt31                |checkpoint    |User     |2020-10-17T20:19:24Z   |00000
|xx.10.0x.xxxxAZ-10-g4c6b4446bd6|
|startup-config        |startup       |User     |2020-10-17T20:47:11Z   |00000
|xx.10.0x.xxxxAZ-10-g4c6b4446bd6|
|ckpt32                |checkpoint    |User     |2020-10-17T20:50:24Z   |00000
|xx.10.0x.xxxxAZ-10-g4c6b4446bd6|
|CPC20171017205110     |checkpoint    |System   |2020-10-17T20:51:10Z   |00000
|xx.10.0x.xxxxAZ-10-g4c6b4446bd6|
```

Showing a detailed list of saved checkpoints for a specific date range:

```
switch# show checkpoint list date 2020-03-08 2020-03-12
|NAME  |TYPE       |WRITER |DATE                |HARDWARE |IMAGE VERSION |
|ckpt2 |checkpoint |User   |2020-03-08 18:10:01 |         |0.0.0         |
|ckpt3 |checkpoint |User   |2020-03-09 23:11:02 |         |0.0.0         |
|ckpt4 |checkpoint |User   |2020-03-11 00:00:03 |         |0.0.0         |
```

write memory

Syntax

`write memory`

Description

Saves the running configuration to the startup configuration. It is an alias of the command `copy running-config startup-config`. If the startup configuration is already present, this command overwrites the startup configuration.

Command context

Manager (`#`)

Authority

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

Examples

```
switch# write memory
Success
```

Boot commands

boot fabric-module

Syntax

`boot fabric-module <SLOT-ID>`

Description

Reboots the specified fabric module.

Command context

Manager (`#`)

Parameters

`<SLOT-ID>`

Specifies the member and slot of the module in the format `member/slot`. For example, to specify the module in member 1 slot 3, enter `1/3`.

Authority

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

Usage

The `boot fabric-module` command reboots the specified fabric module. Traffic performance is affected while the module is down.

If the specified module is the only fabric module in an up state, rebooting that module stops traffic switching between line modules and the line modules power down. The line modules power up when one fabric module returns to an up state.

This command is valid for fabric modules only.

Examples

Rebooting the fabric module in slot **1/3** when auto-confirm is not enabled:

```
switch# boot fabric-module 1/3
This command will reboot the specified fabric module.  Traffic performance may
be affected while the module is down.  Rebooting the last fabric module will
stop traffic switching between line modules.
Do you want to continue (y/n)? y

switch#
```

Rebooting the fabric module in slot **1/1** when auto-confirm is enabled:

```
switch# boot fabric-module 1/3
This command will reboot the specified fabric module.  Traffic performance may
be affected while the module is down.  Rebooting the last fabric module will
stop traffic switching between line modules.
Do you want to continue (y/n) y (auto-confirm)

switch#
```

boot line-module

Syntax

```
boot line-module <SLOT-ID>
```

Description

Reboots the specified line module.

Command context

Manager (#)

Parameters

<SLOT-ID>

Specifies the member and slot of the module in the format `member/slot`. For example, to specify the module in member 1 slot 3, enter `1/3`.

Authority

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

Usage

This command is supported on switches that have multiple line modules.

Reboots the specified line module. Any traffic for the switch passing through the affected module (SSH, TELNET, and SNMP) is interrupted. It can take up to 2 minutes to reboot the module. During that time, you can monitor progress by viewing the event log.

This command is valid for line modules only.

Examples

Reloading the module in slot 1/1:

```
switch# boot line-module 1/1
This command will reboot the specified line module and interfaces on this
module will not send or receive packets while the module is down. Any
traffic passing through the line module will be interrupted. Management
sessions connected through the line module will be affected. It might take
up to 2 minutes to complete rebooting the module. During that time, you can
monitor progress by viewing the event log.
Do you want to continue (y/n)? y
switch#
```

boot management-module

Syntax

```
boot management-module {active | standby | <SLOT-ID>}
```

Description

Reboots the specified management module. Choose the management module to reboot by role (active or standby) or by slot number.

Command context

Manager (#)

Parameters

active

Selects the active management module.

standby

Selects the standby management module.

<SLOT-ID>

Specifies the member and slot of the management module in the format `member/slot`. For example, to specify the module in member 1 slot 5, enter `1/5`.

Authority

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

Usage

This command is supported on switches that have multiple management modules.

This command reboots a single management module in a chassis. Choose the management module to reboot by role (active or standby) or by slot number.

You can use the `show images` command to show information about the primary and secondary system images.

If you reboot the active management module and the standby management module is available, the active management module reboots and the standby management module becomes the active management module.

If you reboot the active management module and the standby management module is not available, you are warned, you are prompted to save the configuration, and you are prompted to confirm the operation.

If you reboot the standby management module, the standby management module reboots and remains the standby management module.

If you attempt to reboot a management module that is not available, the `boot` command is aborted.

Saving the configuration is not required. However, if you attempt to save the configuration and there is an error during the save operation, the `boot` command is aborted.

Hewlett Packard Enterprise recommends that you use the `boot management-module` command instead of pressing the module reset button to reboot a management module because if you are rebooting the only available management module, the `boot management-module` command enables you to save the configuration, cancel the reboot, or both.

Examples

Rebooting the active management module when the standby management module is available:

```
switch# boot management-module active
The management-module in slot 1/5 is going down for reboot now.
```

Rebooting the active management module when the standby management module is not available:

```
switch# boot management-module 1/5
The management module in slot 1/5 is currently active and no
standby management module was found.
```

```
This will reboot the entire switch.

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.
```

boot set-default

Syntax

```
boot set-default {primary | secondary}
```

Description

Sets the default operating system image to use when the system is booted.

Command context

Manager (#)

Parameters

`primary`

Selects the primary network operating system image.

`secondary`

Selects the secondary network operating system image.

Authority

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

Example

Selecting the primary image as the default boot image:

```
switch# boot set-default primary
Default boot image set to primary.
```

boot system

Syntax

```
boot system [primary | secondary | serviceos]
```

Description

Reboots all modules on the switch. By default, the configured default operating system image is used. Optional parameters enable you to specify which system image to use for the reboot operation and for future reboot operations.

Command context

Manager (#)

Parameters

`primary`

Selects the primary operating system image for this reboot and sets the configured default operating system image to `primary` for future reboots.

`secondary`

Selects the secondary operating system image for this reboot and sets the configured default operating system image to `secondary` for future reboots.

`serviceos`

Selects the service operating system for this reboot. Does not change the configured default operating system image. The service operating system acts as a standalone bootloader and recovery OS for switches running the AOS-CX operating system and is used in rare cases when troubleshooting a switch.

Authority

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

Usage

This command reboots the entire system. If you do not select one of the optional parameters, the system reboots from the configured default boot image.

You can use the `show images` command to show information about the primary and secondary system images.

Choosing one of the optional parameters affects the setting for the default boot image:

- If you select the `primary` or `secondary` optional parameter, that image becomes the configured default boot image for future system reboots. The command fails if the switch is not able to set the operating system image to the image you selected.

  You can use the `boot set-default` command to change the configured default operating system image.
- If you select `serviceos` as the optional parameter, the configured default boot image remains the same, and the system reboots all management modules with the service operating system.

If the configuration of the switch has changed since the last reboot, when you execute the `boot system` command you are prompted to save the configuration and you are prompted to confirm the reboot operation.

Saving the configuration is not required. However, if you attempt to save the configuration and there is an error during the save operation, the `boot system` command is aborted.

Examples

Rebooting the system from the configured default operating system image:

```
switch# boot system
Do you want to save the current configuration (y/n)? y
The running configuration was saved to the startup configuration.

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.
```

Rebooting the system from the secondary operating system image, setting the secondary operating system image as the configured default boot image:

```
switch# boot system secondary
Default boot image set to secondary.

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.
```

Canceling a system reboot:

```
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)? n
Reboot aborted.
switch#
```

show boot-history

Syntax

```
show boot-history [all]
```

Description

Shows boot information. When no parameters are specified, shows the most recent information about the boot operation, and the three previous boot operations for the active management module. When the `all` parameter is specified, shows the boot information for the active management module and all available line modules. To view boot-history on the standby, the command must be sent on the standby console.

Command context

Manager (`#`)

Parameters

`all`

Shows boot information for the active management module and all available line modules.

Authority

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

Usage

This command displays the boot-index, boot-ID, and up time in seconds for the current boot. If there is a previous boot, it displays boot-index, boot-ID, reboot time (based on the time zone configured in the system) and reboot reasons. Previous boot information is displayed in reverse chronological order.

`Index`

The position of the boot in the history file. Range: 0 to3.

`Boot ID`

A unique ID for the boot . A system-generated 128-bit string.

Current Boot, up for *<SECONDS>* seconds

For the current boot, the `show boot-history` command shows the number of seconds the module has been running on the current software.

Timestamp boot reason

For previous boot operations, the `show boot-history` command shows the time at which the operation occurred and the reason for the boot. The reason for the boot is one of the following values:

<DAEMON-NAME> `crash`

The daemon identified by *<DAEMON-NAME>* caused the module to boot.

`Kernel crash`

The operating system software associated with the module caused the module to boot.

`Reboot requested through database`

The reboot occurred because of a request made through the CLI or other API.

`Uncontrolled reboot`

The reason for the reboot is not known.

Examples

Showing the boot history of the active management module:

```
switch# show boot-history
Management module
=================

Index : 3
Boot ID : f1bf071bdd04492bbf8439c6e479d612
Current Boot, up for 22 hrs 12 mins 22 secs

Index : 2
Boot ID : edfa2d6598d24e989668306c4a56a06d
07 Aug 18 16:28:01 : Reboot requested through database

Index : 1
Boot ID : 0bda8d0361df4a7e8e3acdc1dba5caad
07 Aug 18 14:08:46 : Reboot requested through database

Index : 0
Boot ID : 23da2b0e26d048d7b3f4b6721b69c110
07 Aug 18 13:00:46 : Reboot requested through database
switch#
```

Showing the boot history of the active management module and all line modules:

```
switch# show boot-history all
Management module
=================

Index : 3
Boot ID : f1bf071bdd04492bbf8439c6e479d612
Current Boot, up for 22 hrs 12 mins 22 secs

Index : 2
Boot ID : edfa2d6598d24e989668306c4a56a06d
07 Aug 18 16:28:01 : Reboot requested through database

Index : 1
Boot ID : 0bda8d0361df4a7e8e3acdc1dba5caad
07 Aug 18 14:08:46 : Reboot requested through database
```

```
Index : 0
Boot ID : 23da2b0e26d048d7b3f4b6721b69c110
07 Aug 18 13:00:46 : Reboot requested through database

Line module 1/1
================
Index : 3
10 Aug 17 12:45:46 : dune_agent crashed
...
```

Firmware management commands

copy {primary | secondary} <*REMOTE-URL*>

Syntax

`copy {primary | secondary} <REMOTE-URL> [vrf <VRF-NAME>]`

Description

Uploads a firmware image to a TFTP or SFTP server.

Command context

Manager (#)

Parameters

`{primary | secondary}`

Selects the primary or secondary image profile to upload. Required

<REMOTE-URL>

Specifies the URL to receive the uploaded firmware using SFTP or TFTP. For information on how to format the remote URL, see URL formatting for copy commands.

`vrf <VRF-NAME>`

Specifies a VRF name. Default: default.

Authority

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

Examples

TFTP upload:

```
switch# copy primary tftp://192.0.2.0/00_10_00_0002.swi
################################################################################ 100.0%
Verifying and writing system firmware...
```

SFTP upload:

```
switch# copy primary sftp://swuser@192.0.2.0/00_10_00_0002.swi
swuser@192.0.2.0's password:
Connected to 192.0.2.0.
sftp> put primary.swi XL_10_00_0002.swi
```

```
Uploading primary.swi to /users/swuser/00_10_00_0002.swi
primary.swi                          100%  179MB  35.8MB/s   00:05
```

copy {primary | secondary} *<FIRMWARE-FILENAME>*

Syntax

`copy {primary | secondary} <FIRMWARE-FILENAME>`

Description

Copies a firmware image to USB storage.

Command context

Manager (`#`)

Parameters

`{primary | secondary}`

Selects the primary or secondary image from which to copy the firmware. Required

`<FIRMWARE-FILENAME>`

Specifies the name of the firmware file to create on the USB storage device. Prefix the filename with `usb:/`. For example: `usb:/firmware_v1.2.3.swi`

For information on how to format the path to a firmware file on a USB drive, see USB URL.

Authority

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

Examples

```
switch# copy primary usb:/11.10.00.0002.swi
```

copy primary secondary

Syntax

`copy primary secondary`

Description

Copies the firmware image from the primary to the secondary location.

Command context

Manager (`#`)

Authority

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

Examples

```
switch# copy primary secondary
The secondary image will be deleted.
```

```
Continue (y/n)? y
Verifying and writing system firmware...
```

copy *<REMOTE-URL>*

Syntax

`copy <REMOTE-URL> {primary | secondary} [vrf <VRF-NAME>]`

Description

Downloads and installs a firmware image from a TFTP or SFTP server.

Command context

Manager (#)

Parameters

<REMOTE-URL>

Specifies the URL from which to download the firmware using SFTP or TFTP. For information on how to format the remote URL, see URL formatting for copy commands.

`{primary | secondary}`

Selects the primary or secondary image profile for receiving the downloaded firmware. Required.

`vrf <VRF-NAME>`

Specifies the name of a VRF. Default: default.

Authority

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

Examples

TFTP download:

```
switch#  copy tftp://192.10.12.0/ss.10.00.0002.swi primary
The primary image will be deleted.

Continue (y/n)? y
######################################################################## 100.0%
Verifying and writing system firmware...
```

SFTP download:

```
switch# copy sftp://swuser@192.10.12.0/ss.10.00.0002.swi primary
The primary image will be deleted.

Continue (y/n)? y
The authenticity of host '192.10.12.0 (192.10.12.0)' can't be established.
ECDSA key fingerprint is SHA256:L64khLwlyLgXlARKRMiwcAAK8oRaQ8C0oWP+PkGBXHY.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.10.12.0' (ECDSA) to the list of known hosts.
swuser@192.10.12.0's password:
Connected to 192.10.12.0.
Fetching /users/swuser/ss.10.00.0002.swi to ss.10.00.0002.swi.dnld
```

```
/users/swuser/ss.10.00.0002.swi                  100%  179MB  25.6MB/s   00:07
Verifying and writing system firmware...
```

copy secondary primary

Syntax

```
copy secondary primary
```

Description

Copies the firmware image from the secondary to the primary location.

Command context

Manager (#)

Authority

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

Examples

```
switch# copy secondary primary
The primary image will be deleted.

Continue (y/n)? y
Verifying and writing system firmware...
```

```
switch# copy sftp://stor@192.22.1.0/im-switch.swi primary vrf mgmt
The primary image will be deleted.

Continue (y/n)? y
The authenticity of host '192.22.1.0 (192.22.1.0)' can't be established.
ECDSA key fingerprint is SHA256:MyI1xbdKnehYut0NLfL69gDpNzCmZqBVvBaRR46m7o8.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.22.1.0' (ECDSA) to the list of known hosts.
stor@192.22.1.0's password:
Connected to 192.22.1.0.
sftp> get c8d5b9f-topflite.swi c8d5b9f-topflite.swi.dnld
Fetching /home/dr/im-switch.swi to c8d5b9f-topflite.swi.dnld
/home/dr/im-switch.swi               100%  226MB  56.6MB/s   00:04

Verifying and writing system firmware...
```

copy *<STORAGE-URL>*

Syntax

```
copy <STORAGE-URL> {primary | secondary}
```

Description

Copies, verifies, and installs a firmware image from a USB storage device connected to the active management module.

Command context

Manager (#)

Parameters

<STORAGE-URL>

Specifies the name of the firmware file to copy from the USB storage device. Required. Prefix the filename with `usb:/`. For example, `usb:/firmware_v1.2.3.swi` For information on how to format the path to a firmware file on a USB drive, see USB URL.

`{primary | secondary}`

Selects the primary or secondary image profile for receiving the copied firmware.

Authority

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

Examples

```
switch# copy usb:/11.10.00.0002.swi primary
The primary image will be deleted.

Continue (y/n)? y

Verifying and writing system firmware...
```

URL formatting for copy commands

TFTP URL

Syntax

`tftp://<IP-ADDR>[:<PORT-NUM>][;blocksize=<Value>]/<FILENAME>`

Examples

To specify a URL with:

- an IPv4 address: `tftp://1.1.1.1/a.txt`
- an IPv6 address: `tftp://[2000::2]/a.txt`
- a hostname: `tftp://hpe.com/a.txt`

To specify TFTP with:

- the port number of the server in the URL: `tftp://1.1.1.1:12/a.txt`
- the blocksize in the URL: `tftp://1.1.1.1;blocksize=1462/a.txt`

  The valid blocksize range is 8 to 65464.
- the port number of the server and blocksize in the URL: `tftp://1.1.1.1:12;blocksize=1462/a.txt`

To specify a file in a directory of URL: `tftp://1.1.1.1/dir/a.txt`

SFTP URL

Syntax

`sftp://<USERNAME>@<IP-ADDR>[:<PORT-NUM>]/<FILENAME>`

Examples

To specify:

- A URL with an IPv4 address: `sftp://user@1.1.1.1/a.txt`
- A URL with an IPv6 address: `sftp://user@[2000::2]/a.txt`
- A URL with a hostname: `sftp://user@hpe.com/a.txt`
- SFTP port number of a server in the URL: `sftp://user@1.1.1.1:12/a.txt`
- A file in a directory of URL: `sftp://user@1.1.1.1/dir/a.txt`
- To specify a file with absolute path in the URL: `sftp://user@1.1.1.1//home/user/a.txt`

USB URL

Syntax

`usb:/<FILENAME>`

Examples

To specify a file:

- In a USB storage device: `usb:/a.txt`
- In a directory of a USB storage device: `usb:/dir/a.txt`

Dynamic Segmentation (DS) is an enterprise network solution that combines Aruba OS-CX security and networking features to dynamically place clients into network segments based on client credentials. The client network segments are dynamically carved out of the enterprise networks when on-boarding secure clients. Two options are available:

- User Based Tunnels (UBT).
- Virtual Network Based Tunnels (VNBT). Also called switch-to-switch dynamic segmentation.

In both solutions, once authenticated (using MAC-Auth or 802.1X) an enterprise client is bound to a network role and a VLAN is associated with the role. User traffic is then placed on the VLAN (know as the role VLAN) corresponding to the role to which the user belongs. Role association is defined using the individual client authentication mode or using device-profile based authentication.

The administrator must pre-configure all potential role VLANs and VRFs in all access switches (and additional configuration such as IGMP snooping on VLAN, PIM RP, etc.). The switch ensures that the role VLANs and VRFs are instantiated only upon client on-boarding on the target VLAN (using the command `system vlan-client-presence-detect`). This ensures that unnecessary broadcast domain creations and route learning do not occur.

Virtual network based tunneling

Segment definition

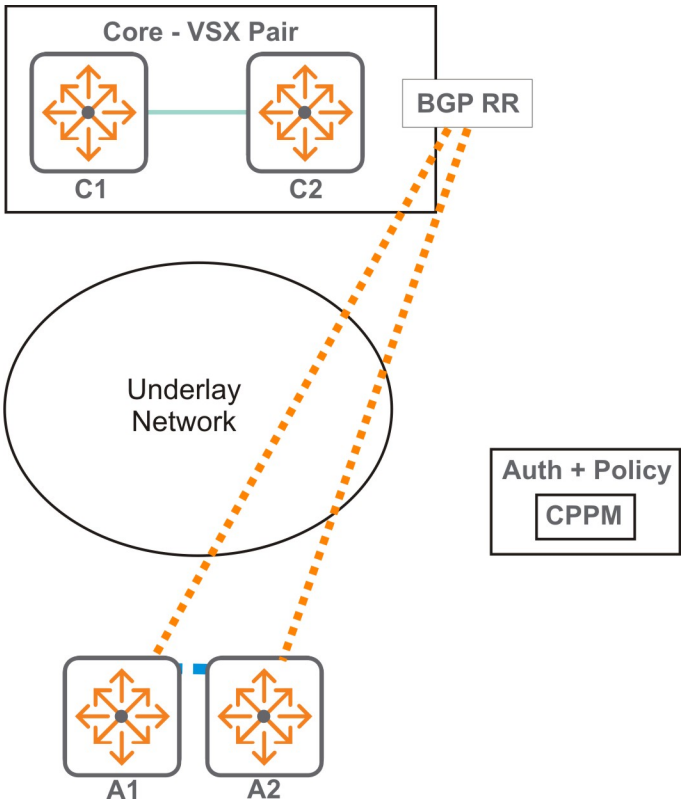
Each segment has its own policies. For example, if a group of clients belong to an *admin* segment, then this segment can have better QoS and security privileges as compared to the segments assigned to guest clients.

Inter-segment traffic is prohibited between two segments based on policy.

A segment does not map to a network construct such as a VLAN or a VRF. Multiple segments can co-exist within a VLAN or a segment can span multiple VLANs and VRFs. However, the switch must realize segmentation using network constructs such as VLANs, VRFs ACLs, etc.

Example

This example illustrates a simple deployment using two VLANs and VRFs.



Overlay Client VLAN	L2 VLAN	Subnet
10	100	1.1.1.0/24
20	200	2.2.2.0/24

Overlay Client VRF	L3 VNI	Overlay SVIs on VRF	Overlay ROPs on VRF
A	10000	▪ VLAN interface of 10, 20 on access. ▪ VLAN interface to the north of Core (if any needed).	ROP interfaces to the north of Core.

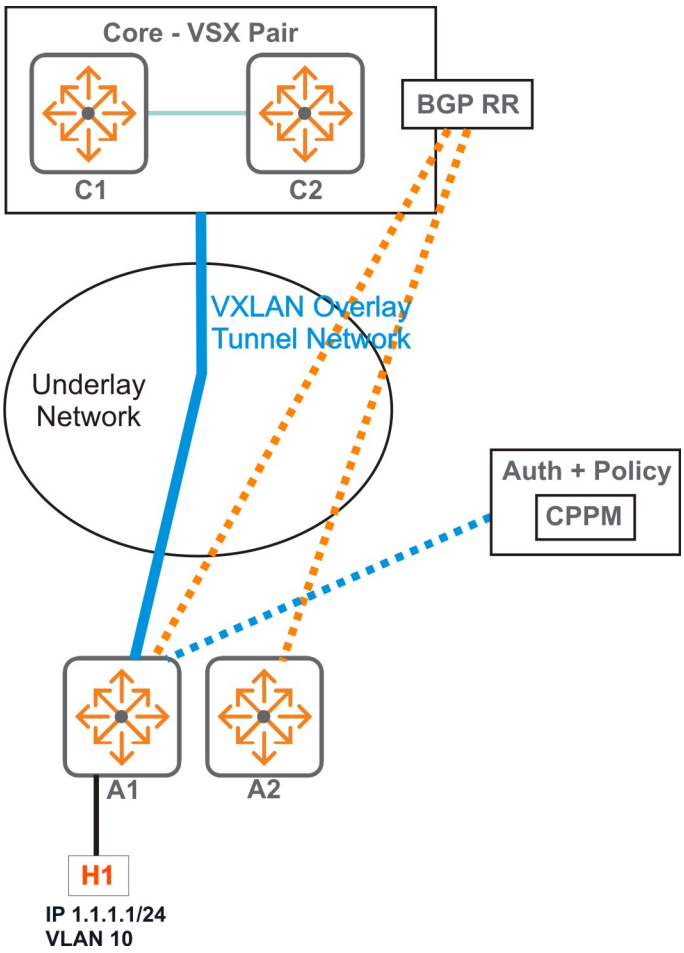
Configuration on switches	A1	A2	Core
▪ L2 VNI 100. ▪ Anycast Gateway for 1.1.1.0/24. ▪ VLAN interface for 10.	Y	Y	N
▪ L2 VNI 200. ▪ Anycast Gateway for 2.2.2.0/24. ▪ VLAN interface for 20.	Y	Y	N
▪ VRF A.	Y	Y	Y

Configuration on switches	A1	A2	Core
▪ L3 VRF for A.			
▪ RD, RTs for VRF A. (Can be derived from L3 VNI too.)	Y	Y	Y

The two VRFs are configured on the core switch, and the two VLANs and VRFs are configured on the two access switches as required. The two VLANs and the VRF are part of the *running config* on both access switches.

Initially, the status of the two VLANs on the access switches is *down*. This means that:

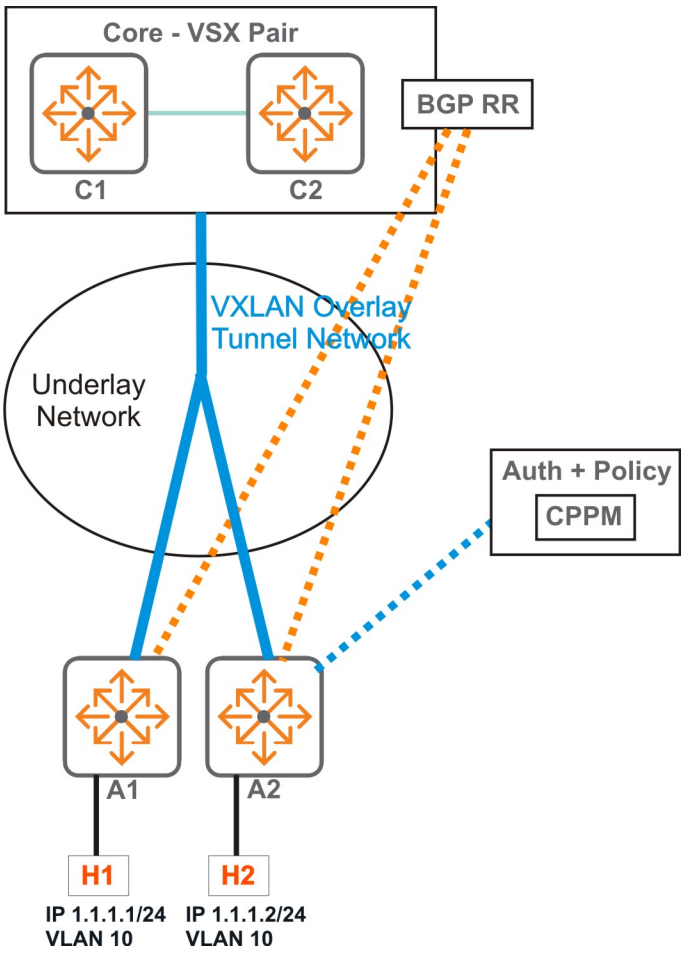
- EVPN routes - RT-3 (IMET) route, and RT-5 and RT-2 with respect to the VLAN interfaces are not announced by the switches.
- No VXLAN tunnels are established between any pairs of switches.



When Host H1 connects to A1, the host is authenticated with CPPM and the client is mapped to Role-1 on VLAN 10. This results in the following:

- The VLAN state changes to *up* in show commands on A1.
- The L2 and L3 forwarding constructs for the local MAC of H1 are programmed onto VLAN 10 inside A1.

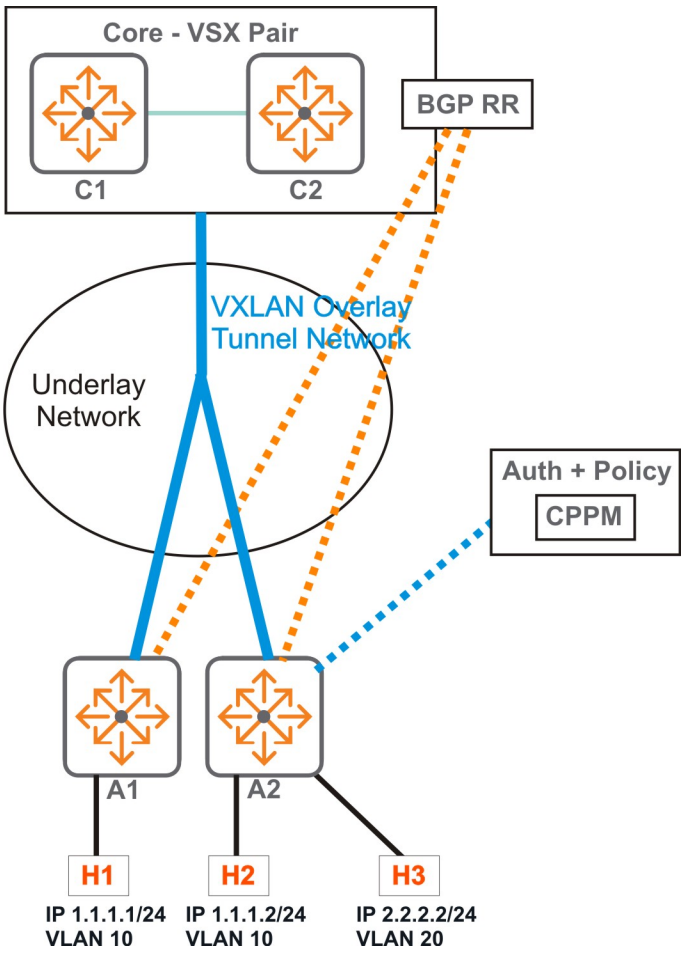
- The IMET route for L2VNI is advertised by A1.
  - This route is not used by the core as it does not have footprint of VLAN 10 on it.
  - This route is not used by A2 either - this is because it does not have a local VLAN 10 on "up" state as yet.
- RT-5 prefix route (if enabled) is advertised by A1.
  - Upon receiving the route, the core programs the prefix route (1.1.1.0/24). This also results in VxLAN tunnel programming on the core towards A1.
  - A2 still does not use this route, because the VRF is not instantiated on A2 yet.
- The RT-4 and the RT-2 routes are advertised by A1.
  - Upon receiving the route, the core programs the route host route (1.1.1.1/32) with BH as A1. But the existing tunnel towards A1 is reused.
  - A2 still does not use this route.
- The BUM domain for VLAN 10 on A1 is still the local host H1. This is because VLAN 10 is not instantiated on any other switch as yet.
- Any prefix routes from the core is programmed by A1 and it also programs a VxLAN tunnel towards the core.



When Host H2 connects to A2, the host is authenticated with CPPM and the client is mapped to say Role-1 and the role's VLAN is VLAN 10. This results in the following:

- The VLAN state is changes to "up" in the show commands.
- The L2 and L3 forwarding constructs for the local MAC of H2 is programmed into VLAN 10 inside A2.

- The IMET route for L2VNI is advertised by A2.
  - This route is not used by the core again as it does not have footprint of VLAN 10 on it.
  - This route is used by A1. It creates a VXLAN tunnel towards A2 and adds it to the BUM domain for VLAN 10.
- RT-5 prefix route (if enabled) is advertised by A2.
  - Upon receiving the route, the core programs the prefix route (1.1.1.0/24). In the absence of ECMP, the existing route in the core is overwritten. This also results in VxLAN tunnel programming on the core towards A2.
  - A1 still does not use this route, this is because the local connected route for 1.1.1.0/24 has higher priority.
- The RT-4 and the RT-2 routes are advertised by A2.
  - Upon receiving the route, the core programs the route host route (1.1.1.2/32). The existing tunnel is reused.
  - A1 programs the 1.1.1.2/32 route into its FIB with NH as VTEP towards A2.
- Any prefix routes from the core are programmed by A2 and it also programs a VxLAN tunnel towards the core
- Thus a full mesh of VxLAN tunnels is created.



If host H3 connects to A2 and it is on boarded on VLAN 20. This results in the following:

- The VLAN state of VLAN 20 changes to "up" in the show commands
- The L2 and L3 forwarding constructs for the local MAC of H3 is programmed into VLAN 20 inside A2.

- The IMET route for L2VNI is advertised by A2
  - This route is not used by the core again as it does not have footprint of VLAN 20 on it.
  - This route is not used by A1 either for the same reason.
- RT-5 prefix route (if enabled) is advertised by A2
  - Upon receiving the route, the core programs the prefix route (2.2.2.0/24). The core reuses the existing tunnel.
  - A1 also programs the prefix route with NH as A2.
- The RT-4 and the RT-2 routes with respect to H3 are advertised by A2.
  - A1 programs its FIB for host route 2.2.2.2/24 with NH as A2. But it reuses the existing tunnel.
  - Same behavior on the core.

Additional notes

- Reference counts maintained in the access switches ensure that existing tunnels are reused as and when new clients come up.
- The clients leaving the VLAN (disconnect/Auth time out etc.) can lead to the reversal of the procedure described above - i.e. deletion of local programming, withdrawal of routes, VLAN status change, etc. The reversal is initiated based on reference count.
- Dynamic VLAN instantiation does not mandate VNI association for VLANs. Even local VLANs with secure clients (if any) are also dynamically instantiated.

User-based tunneling

User-based tunneling uses GRE to tunnel ingress traffic on a switch interface to a mobility controller for further processing. User-based tunneling enables a mobility controller to provide a centralized security policy, using per-user authentication and access control to ensure consistent access and permissions.

Applications of user-based tunneling include:

- Traffic segmentation: Enables splitting of traffic based on user credentials, rather than the physical port to which a user is connected. For example, guests on a corporate network can be assigned to a specific VLAN with access and firewall policies defined to protect the network. Traffic from computers/laptops can be tunneled, while allowing VoIP traffic to move freely through the wired network.
- Authentication of PoE devices: Many devices that require power over Ethernet (PoE) and network access, such as security cameras, payment card readers, and medical devices, do not have built-in security software. As a result, these devices can pose a risk to networks. User-based tunneling can authenticate these devices and tunnel their traffic to a mobility controller, harnessing the firewall and policy capabilities to secure the network.

At the most basic level User-Based Tunneling has two components:

- **User-Roles** refers to the ability to assign roles, on the fly, to a wired device/user, based on such things as the access method of a client. When leveraging ClearPass, additional context can be added, such as time-of-day and type-of-machine. As a result, IT staff no longer must pre-configure an access-port to VLAN and uplinks.
- **Tunneling** is the ability to tunnel traffic back to an Aruba Mobility Controller (previously known as tunneled-node).

User-based tunneling supports two types of controller deployments:

- Standalone Controller Support
- Clustered Controller Support

The recommended controller version for user-based tunneling is 8.5 or greater.

Components of user-based tunneling

Clients and devices

Traditionally, ports were labeled with a color and a color was assigned to a specific device. With colorless ports, all ports on an access switch are set to authenticate with both 802.1X and MAC Authentication. When a device connects to the network it is authenticated using either MAC Authentication or 802.1X and triggers an enforcement policy from ClearPass, which contains an enforcement profile with a user role configuration.

Access switches

Access switches authenticate users connected to the switch. Once a device or user is authenticated, a role is applied to the device or user. A role is a set of attributes and policies that is applied to the device or user. This user role can exist locally on an access switch or on ClearPass as part of an enforcement profile.

Mobility controller cluster

The Aruba Mobility Controller has many built-in security and application capabilities tailored specifically to wireless traffic. However, this can be extended as well to wired traffic. This is the main reason to tunnel traffic from an Aruba access switch to a controller, so the wired, tunneled traffic can take advantage of the controller's firewall capabilities and client applications.

Aruba ClearPass Policy Manager

ClearPass assigns enforcement policies and profiles containing user role information based on profiled devices or authenticated user information.

How it works

When first configuring the switch, the *tunneling profile* must be configured first. This is done using the command `ubt zone`. Within this context, the primary controller IP address can be configured, which should be the physical IP of one of the cluster members. Once the controller information is known on the switch and the UBT service is enabled, the switch then performs a handshake with the controller to determine its reachability and to discover the version information.

When reachability is confirmed, the switch executes a switch bootstrap, and sends a bootstrap message to the controller, similar to an AP Hello between an AP and a controller. This bootstrap control packet contains user role information. Once the controller receives the message, it replies with an acknowledge message. When acknowledged, the switch updates its local data structures with a bucket map and controller node list, which is used for mapping users to controllers and client load balancing.

After the bucket map list is downloaded to the switch, a GRE heartbeat is then started between the switch and the controller, forming a tunnel. A regular heartbeat, using GRE, is exchanged with the controller, which then serves as the switch anchor controller (SAC). This is the `primary-controller ip` in the `ubt zone` command. A secondary heartbeat is also established with a standby controller, acting as a secondary switch anchor controller (s-SAC).

When a user connects to a secure port, the authentication sub-system on the switch sends a RADIUS request to the RADIUS server (for example, ClearPass Policy Manager), which authenticates the user and returns a user role to the switch in the form of a local user role (LUR), downloadable user role (DUR) or venfor-specific attribute (VSA).

For a downloadable user role, the entire role itself is downloaded to the switch containing the user role via a VSA.

Aruba utilizes the concept of a user role which contains user policy and access to the network based on the role. A user-role can contain ACL/QoS policy, captive portal, VLAN information (used for locally switched traffic), and device attributes. When the user role VSA, received from the RADIUS server, is applied to the user, a command to redirect traffic to a controller can be included within the user role. This is defined with the `gateway zone` command which causes tunneling to be enabled. The authentication sub-system notifies the tunneling subsystem on the switch, providing a gateway or secondary role. The gateway or secondary role is the user role on the controller where policy will generally exist for tunneled users, and where firewall and security policies are applied. This can also be the same role used for wireless users and can be reused for wired users, if feasible.

The gateway role information sent to the switch tunneling subsystem is an indication to the controller that it has to enforce additional policies on the user's traffic based on the policy configuration associated with the secondary role and the tunnel. This secondary role can be downloaded directly to the controller. When the primary controller or cluster is not reachable, the SAC tunnel is formed with the backup controller and the clients are tunneled to the backup.

Multi-zoning in UBT

The multizone feature allows a switch to host users whose traffic is tunneled to different gateway clusters. Each UBT zone corresponds to a unique Gateway cluster or a standalone controller. Each UBT zone is mapped to a single VRF on a switch/stack. The zone configuration is local to the switch and it provides isolation of user traffic across different gateway clusters.

Multizone is supported for both UBT modes (local VLAN and VLAN extend). The maximum number of UBT zones that can be configured is eight per switch/stack. All UBT zones configured on a switch/stack will either be in local VLAN or VLAN extend mode. Mixed mode is not supported across zones on the same switch/stack. Multiple UBT zones per single VRF on a switch/stack is not supported.

Points to remember

- **UBT Mode: Local VLAN**
  - UBT is supported only on the default VRF.
  - Source interface is specified using command `ip source-interface`.
  - VLAN is specified using command `ubt-client-vlan`.
  - No feature should be configured on `ubt-client-vlan`.
  - Client IP tracker is not supported for UBT clients.
  - UBT does not support tagged clients.
  - UBT clients and non-UBT clients on same VLAN and same port is not supported.
  - Source interface change: Disable UBT, change the source-interface, enable UBT.
- **UBT Mode: VLAN extend**
  - UBT is supported only on the default VRF.
  - Source interface is specified using command `ip source-interface`.
  - UBT client vlan is defined under role.
  - DHCP snooping and ND-snooping should not enabled on a UBT client assigned VLAN.
  - IGMP snooping should not be enabled on a UBT client assigned VLAN.
  - Client IP tracker is not supported for UBT clients.
  - The VLAN on which UBT clients are placed should not be configured on the switch uplink.
  - UBT clients and non-UBT clients on the same VLAN on the same switch is not supported.
  - Source interface change: Disable UBT, change the source-interface, enable UBT.

- **Gateway DUR**
  - Downloadable gateway role is supported with switch VSA only. Mixed mode role configuration is not supported (switch DUR + gateway DUR) or any other switch plus gateway role combination.
  - CPPM server FQDN/hostname configuration is supported.
- **Multi-zone**
  - Total number of UBT zones supported on switch/stack is 8.
  - Only one UBT zone per VRF is supported. Multiple UBT zones per single VRF is not supported.
  - The total number of supported UBT users across different zones on a switch/stack is 1017.
  - Overlapping user role VLAN(s) should not be present across UBT zones on a switch.

  This will lead to one zone traffic (multicast and broadcast) reaching other zone(s) on the same switch.
- **PC behind an IP phone**

  You should not have a PC and phone on the same VLAN on the same port when the PC is a UBT client and the phone is a non-UBT client. If you do, UBT clients broadcast/multicast packets will return to the same port and corrupt the phone MAC table.
- **Clients behind an L2 switch on the same VLAN**

  You should not have clients behind an L2 switch in a UBT environment. If UBT and non-UBT clients are behind an L2 switch on the same VLAN, this will cause duplicate packets. Broadcast/multicast packets will be copied to the tunnel and locally, causing the client to receive duplicate packets and network instability.

It's recommended to connect the client directly into switch ports or behind VoIP.

Comparison between UBT modes

UBT Mode: Local VLAN	UBT Mode: VLAN extend
Switch unaware of UBT user VLAN	Switch is aware of user VLAN
Colorless ports: No UBT user VLAN config at switch	Colorless ports: UBT User VLAN config is required on switch
VLAN assignment by controller	VLAN assignment by switch
Supports only untagged UBT users	Supports both tagged/untagged UBT users
Controller replicates the broadcast/multicast traffic (converting bcast/mcast to unicast) and sends it to every UBT client	Single dedicated multicast GRE tunnel will be established between the controller and switch for Broadcast/Multicast traffic
Controller forwards all broadcast/multicast traffic to the UBT clients which are part of the same VLAN	Switch will forward the broadcast/multicast traffic to the UBT clients which are part of the same VLAN
The unicast/multicast/broadcast traffic from controller to switch is sent to the clients via the same UAC tunnel	The unicast traffic from controller to switch is sent through the UAC tunnel and multicast/broadcast traffic via Multicast GRE tunnel
Multicast traffic will be sent to the client which send join	Multicast traffic will be sent to all UBT clients on the same VLAN

User-based tunneling commands

backup-controller ip

Syntax

`backup-controller ip <IP-ADDR>`

`no backup-controller ip <IP-ADDR>`

Description

Specifies the IP address of the backup controller for the UBT zone.

The no form of this command deletes the IP address of the backup controller.

Command context

`config-ubt-<ZONE-NAME>`

Parameters

<IP-ADDR>

Specifies the IP address of the backup controller.

Authority

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

Examples

Specifying the backup controller ip address for `zone1`:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# backup-controller ip 10.116.51.11
```

Delete the configured backup controller IP address:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# no backup-controller ip 10.116.51.11
```

enable

Syntax

`enable`

`no enable`

Description

Enables the UBT zone.

The `no` form of this command disables the UBT zone.

Command context

`config-ubt-<ZONE-NAME>`

Authority

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

Examples

Enabling UBT for zone `zone1`:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# enable
```

Disabling UBT for `zone1`:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# no enable
```

ip source-interface

Syntax

```
ip source-interface {all | ubt} {interface <IFNAME> | <IPV4-ADDR>} [vrf <VRF-NAME>]
no ip source-interface {all | ubt} {interface <IFNAME> | <IPV4-ADDR>} [vrf <VRF-NAME>]
```

Description

Sets a single source IP address for the UBT zone VRF. This ensures that all traffic sent by UBT zone/VRF has the same source IP address, regardless of how it egresses the switch.

This command provides two ways to set the source IP addresses: either by specifying a static IP address, or by using the address assigned to a switch interface. If you define both options, then the static IP address takes precedence.

The `no` form of this command deletes the single source IP address for UBT.

Command context

`config`

Parameters

`all`

When used no other parameters are required.

`interface <IFNAME>`

Specifies the name of the interface from which UBT obtains its source IP address. The interface must have a valid IP address assigned to it. If the interface has both a primary and secondary IP address, the primary IP address is used.

`<IPV4-ADDR>`

Specifies the source IP address to use for UBT. The IP address must be defined on the switch, and it must exist on the specified VRF, Default: default. Specify the address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255.

`vrf <VRF-NAME>`

Specifies the name of the VRF from which the UBT zone sets its source IP address.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Setting interface `1/1/7` as the source address for UBT for VRF `default`:

```
switch(config)# ip source-interface ubt interface 1/1/7 vrf default
```

Deleting the configured source interface `1/1/7` as the source address for UBT for VRF `default`:

```
switch(config)# no ip source-interface ubt interface 1/1/7 vrf default
```

Specifying the static IP address `1.1.1.1` as the source address for UBT for VRF `default`:

```
switch(config)# ip source-interface ubt 1.1.1.1 vrf default
```

Deleting the configured ip address as the source address for UBT for VRF `default`:

```
switch(config)# no ip source-interface ubt 1.1.1.1 vrf default
```

papi-security-key

Syntax

```
papi-security-key [{ciphertext <SEC-KEY> | plaintext <SEC-KEY>}]
no papi-security-key
```

Description

Specifies the shared security key used to encrypt UBT PAPI messages exchanged between the switch and the controller cluster for the zone.

The `no` form of this command deletes the shared security key .

Command context

`config-ubt-<ZONE-NAME>`

Parameters

`ciphertext <SEC-KEY>`

Specifies an encrypted security key.

`plaintext <SEC-KEY>`

Specifies a plaintext security key. Range: 10 to 64 characters.

When the security key is not provided on the command line, plaintext security key prompting occurs upon pressing Enter. The entered security key characters are masked with asterisks..

Authority

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

Examples

Specifying the PAPI security key for UBT zone `zone1` as plaintext:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# papi-security-key plaintext F82#450b
```

Specifying the PAPI security key for UBT `zone2` with plaintext prompting:

```
switch(config)# ubt zone zone2
switch(config-ubt-zone2)# papi-security-key
Enter the PAPI security key: **********
Re-Enter the PAPI security key: **********
```

Specifying the PAPI security key for UBT `zone1` as ciphertext:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# papi-security-key ciphertext AQBapdAVz5...RmH3+4cpg=
```

Removing the PAPI security key for UBT `zone1`:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# no papi-security-key
```

primary-controller ip

Syntax

`primary-controller ip <IP-ADDR>`

`no primary-controller ip <IP-ADDR>`

Description

Specifies the IP address of the primary controller IP address for the zone.

The `no` form of this command deletes the IP address of the primary controller.

Command context

`config-ubt-<ZONE-NAME>`

Parameters

<IP-ADDR>

Specifies the IP address of the primary controller.

Authority

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

Examples

Specify the primary controller IP address for `zone1`:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# primary-controller ip 10.116.51.10
```

Delete the configured primary controller IP address:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# no primary-controller ip 10.116.51.10
```

sac-heartbeat-interval

Syntax

`sac-heartbeat-interval <TIME>`

`no sac-heartbeat-interval <TIME>`

Description

Specifies the SAC heartbeat refresh time interval in seconds.

The `no` form of this command sets the heartbeat interval to the default value.

Command context

`config-ubt-<ZONE-NAME>`

Parameters

<TIME>

Specifies the SAC heartbeat refresh time interval in seconds. Range: 1 to 8. Default: 1.

Authority

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

Examples

Specifying a heartbeat refresh interval of 1 for UBT `zone1`:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# sac-heartbeat-interval 1
```

Deleting the configured heartbeat refresh interval:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# no sac-heartbeat-interval
```

show ip source-interface ubt

Syntax

`show ip source-interface ubt`

Description

Displays source IP address configuration information for the UBT zone(s).

Command context

Operator (`>`) or Manager (`#`)

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Showing source IP address configuration information:

```
switch(config)# show ip source-interface ubt

Source-interface Configuration Information
------------------------------------------------------------------------
Protocol        Src-Interface       Src-IP                  VRF
------------------------------------------------------------------------
ubt             vlan10              10.1.1.2                default
ubt             vlan20              20.1.1.2                blue
```

show capacities ubt

Syntax

```
show capacities ubt
```

Description

Shows the maximum number of UBT clients and zones which can be configured in the system.

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 maximum number of UBT clients and zones which can be configured:

```
switch# show capacities ubt

System Capacities: Filter UBT
Capacities Name                                                                 Value
--------------------------------------------------------------------------------------
Maximum number of UBT clients in a system                                        1017
Maximum number of UBT zones per VRF                                                 1
Maximum number of UBT zones                                                         8
```

show ubt

Syntax

```
show ubt [brief]
show ubt zone <ZONE-NAME> [brief]
```

Description

Shows global configuration information for UBT in addition to detailed or brief information for a specific UBT zone.

Command context

Operator (>) or Manager (#)

Parameters

`zone <ZONE-NAME>`

Specifies the name of a zone. Length: 1 to 64 characters.

`brief`

Displays brief information.

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 global UBT configuration information where local-VLAN mode has been configured:

```
switch# show ubt

Zone Name                 : zone1
UBT Mode                  : local-vlan
Primary Controller        : 10.116.51.10
Backup Controller         : 10.116.51.11
SAC HeartBeat Interval    : 1
UAC KeepAlive Interval    : 60
Reserved VLAN Identifier  : 4094
VRF Name                  : default
Admin State               : ENABLED
PAPI Security Key         : AQBapdxySvGPvdTl ... bL4FE=

Zone Name                 : zone2
UBT Mode                  : local-vlan
Primary Controller        : 1.1.5.10
Backup Controller         : 1.1.5.11
SAC HeartBeat Interval    : 1
UAC KeepAlive Interval    : 60
Reserved VLAN Identifier  : 4094
VRF Name                  : blue
Admin State               : ENABLED
PAPI Security Key         : TRQapdxySvGPvdTlkYn1 ... zP4FE=
```

Showing global UBT configuration information where VLAN-extend mode has been configured:

```
switch# show ubt

Zone Name                 : zone1
UBT Mode                  : vlan-extend
Primary Controller        : 10.116.51.10
Backup Controller         : 10.116.51.11
SAC HeartBeat Interval    : 1
UAC KeepAlive Interval    : 60
Reserved VLAN Identifier  : -NA-
VRF Name                  : default
Admin State               : ENABLED
PAPI Security Key         : AQBapdxySvGPvdTlkYn1 ... bL4FE=

Zone Name                 : zone2
```

```
UBT Mode                  : vlan-extend
Primary Controller        : 1.1.5.10
Backup Controller         : 1.1.5.11
SAC HeartBeat Interval    : 1
UAC KeepAlive Interval    : 60
Reserved VLAN Identifier  : -NA-
VRF Name                  : blue
Admin State               : ENABLED
PAPI Security Key         : TRQapdxySvGPvdTlkYn1 ... zP4FE=
```

Showing brief global UBT configuration information where local-VLAN mode has been configured:

```
switch(config)# show ubt brief
-------------------------------------------------------------------------------
Zone Name    UBT Mode     Primary Controller Address    VRF Name    Status
-------------------------------------------------------------------------------
zone1        local-vlan   10.116.51.10                  default      Enabled
zone2        local-vlan   1.1.5.10                      blue         Enabled
```

Showing brief global UBT configuration information where VLAN-extend mode has been configured:

```
switch# show ubt brief
-------------------------------------------------------------------------------
Zone Name    UBT Mode         Primary Controller Address    VRF Name     Status
-------------------------------------------------------------------------------
zone1        vlan-extend      10.116.51.10                  default      Enabled
zone2        vlan-extend      1.1.5.10                      blue         Enabled
```

Showing brief configuration for UBT `zone1` where local-VLAN mode has been configured:

```
switch# show ubt zone zone1 brief
-------------------------------------------------------------------------------
Zone Name    UBT Mode         Primary Controller Address    VRF Name     Status
-------------------------------------------------------------------------------
zone1        local-vlan       30.116.51.10                  default      Enabled
```

Showing brief configuration for UBT `zone1` where VLAN-extend mode has been configured:

```
switch# show ubt zone zone1 brief
-------------------------------------------------------------------------------
Zone Name    UBT Mode         Primary Controller Address    VRF Name     Status
-------------------------------------------------------------------------------
zone1        vlan-extend      10.116.51.10                  default      Enabled
```

show ubt information

Syntax

```
show ubt information
show ubt information zone <ZONE-NAME>
```

Description

Shows SAC and UAC information for UBT. Specifying a zone name displays UBT information for that zone.

Command context

Operator (>) or Manager (#)

Parameters

ZONE-NAME

Specifies UBT zone name. Maximum characters: 64.

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 SAC and UAC information for the tunneled node server:

```
switch(config)# show ubt information

===============================================================================
Zone zone1:
===============================================================================
SAC Information :

  Active         : 10.1.1.2
  Standby        : 10.1.1.3

Node List Information :

  Cluster Name        : cluster1

  Cluster Alias Name  :

  Node List      :
  ---------------
    10.1.1.2
    10.1.1.3
    10.1.1.4

Bucket Map Information :

Bucket Map Active   : [0...255]

Bucket ID  A-UAC           S-UAC               Connectivity
-------------------------------------------------------------
0          10.1.1.2        10.1.1.3            L2
1          10.1.1.3        10.1.1.4            L2
2          10.1.1.4        10.1.1.2            L2
...

===============================================================================
Zone zone2:
===============================================================================
SAC Information :

  Active         : 20.1.1.2
  Standby        : 20.1.1.3

Node List Information :
```

```
  Cluster Name        : cluster2

  Cluster Alias Name  :

  Node List      :
  ----------------
    20.1.1.2
    20.1.1.3
    20.1.1.4

Bucket Map Information :

Bucket Map Active   : [0...255]

Bucket ID  A-UAC           S-UAC               Connectivity
-------------------------------------------------------------
0          20.1.1.2        20.1.1.3            L2
1          20.1.1.3        20.1.1.4            L2
2          20.1.1.4        20.1.1.2            L2
...
```

Showing SAC and UAC information for `zone1`:

```
switch(config)# show ubt information zone zone1

==========================================================================
Zone zone1:
==========================================================================
SAC Information :

  Active         : 10.1.1.2
  Standby        : 10.1.1.3

Node List Information :

  Cluster Name        : cluster1

  Cluster Alias Name  :

  Node List      :
  ----------------
    10.1.1.2
    10.1.1.3
    10.1.1.4

Bucket Map Information :

Bucket Map Active   : [0...255]

Bucket ID  A-UAC           S-UAC               Connectivity
-------------------------------------------------------------
0          10.1.1.2        10.1.1.3            L2
1          10.1.1.3        10.1.1.4            L2
2          10.1.1.4        10.1.1.2            L2
...
```

show ubt state

Syntax

```
show ubt state
show ubt state zone <ZONE-NAME>
show ubt state zone <ZONE-NAME> uac-ip <UAC-ADDR>
```

Description

Shows the global UBT state.

Specifying a zone shows the UBT state of that zone.

Specifying a UAC IP address shows the UBT state of that UAC.

Command context

Operator (`>`) or Manager (`#`)

Parameters

`zone <ZONE-NAME>`

Specifies UBT zone name. Maximum characters: 64.

`uac-ip <UAC-ADDR>`

Specifies the IP address of the user anchor controller for which to view user information. Specify the address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Showing the UBT state where local-VLAN mode has been configured:

```
switch# show ubt state
================================================================================
Zone zone1:
================================================================================
Local Conductor Server (LCS) State:
LCS Type      IP Address    State               Role
--------------------------------------------------------------------------------
Primary     : 10.1.1.2      ready_for_bootstrap operational_primary
Secondary   : 10.1.1.10     ready_for_bootstrap operational_secondary
Switch Anchor Controller (SAC) State:
              IP Address      MAC Address          State
----------------------------------------------------------------
Active      : 10.1.1.2        00:0b:86:b7:62:9f    registered
Standby     : 10.1.1.3        00:0b:86:b7:64:0f    registered
User Anchor Controller(UAC): 10.1.1.2
User               Port    State                          Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------------
00:00:00:00:00:01  1/1/1   registered                     5          13       4094
User Anchor Controller(UAC): 10.1.1.3
User               Port    State                          Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------------
00:00:00:00:00:02  1/1/2   registered                     4          14       4094
================================================================================
Zone zone2:
================================================================================
Local Conductor Server (LCS) State:
LCS Type      IP Address    State               Role
```

```
-------------------------------------------------------------------
Primary     : 20.1.1.2      ready_for_bootstrap operational_primary
Secondary   : 20.1.1.10     ready_for_bootstrap operational_secondary
Switch Anchor Controller (SAC) State:
              IP Address      MAC Address          State
-----------------------------------------------------------------
Active      : 20.1.1.2        00:0b:86:b7:62:9f    registered
Standby     : 20.1.1.3        00:0b:86:b7:64:0f    registered
User Anchor Controller(UAC): 20.1.1.2
User               Port    State                          Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------
00:00:00:00:00:03  1/1/1   registered                     5          13       4094
User Anchor Controller(UAC): 20.1.1.3
User               Port    State                          Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------
00:00:00:00:00:04  1/1/2   registered                     4          14       4094
```

Showing the UBT state where VLAN-extend mode has been configured:

```
switch# show ubt state
===================================================================
Zone zone1:
===================================================================
Local Conductor Server (LCS) State:
LCS Type      IP Address    State               Role
-------------------------------------------------------------------
Primary     : 10.1.1.2      ready_for_bootstrap operational_primary

Secondary   : 10.1.1.10     ready_for_bootstrap operational_secondary
Switch Anchor Controller (SAC) State:
              IP Address      MAC Address          State
-----------------------------------------------------------------
Active      : 10.1.1.2        00:0b:86:b7:62:9f    registered
Standby     : 10.1.1.3        00:0b:86:b7:64:0f    registered
User Anchor Controller(UAC): 10.1.1.2
User               Port    State                          Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------
00:00:00:00:00:01  1/1/1   registered                     5          13       10
User Anchor Controller(UAC): 10.1.1.3
User               Port    State                          Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------
00:00:00:00:00:02  1/1/2   registered                     4          14       20
===================================================================
Zone zone2:
===================================================================
Local Conductor Server (LCS) State:
LCS Type      IP Address    State               Role
-------------------------------------------------------------------
Primary     : 20.1.1.2      ready_for_bootstrap operational_primary
Secondary   : 20.1.1.10     ready_for_bootstrap operational_secondary
Switch Anchor Controller (SAC) State:
              IP Address      MAC Address          State
-----------------------------------------------------------------
Active      : 20.1.1.2        00:0b:86:b7:62:9f    registered
Standby     : 20.1.1.3        00:0b:86:b7:64:0f    registered
User Anchor Controller(UAC): 20.1.1.2
User               Port    State                          Bucket ID  Gre Key  VLAN
```

```
-------------------------------------------------------------------------------------
00:00:00:00:00:03  1/1/1   registered                    5          13       30
User Anchor Controller(UAC): 20.1.1.3
User               Port    State                        Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------------
00:00:00:00:00:04  1/1/2   registered                    4          14       40
```

Showing the UBT state of `zone1`:

```
switch# show ubt state zone zone1

==========================================================================
Zone zone1:
==========================================================================
Local Conductor Server (LCS) State:

LCS Type      IP Address    State               Role
--------------------------------------------------------------------------
Primary     : 10.1.1.2      ready_for_bootstrap operational_primary
Secondary   : 10.1.1.10     ready_for_bootstrap operational_secondary

Switch Anchor Controller (SAC) State:

              IP Address      MAC Address          State
--------------------------------------------------------------------
Active      : 10.1.1.2        00:0b:86:b7:62:9f    registered
Standby     : 10.1.1.3        00:0b:86:b7:64:0f    registered

User Anchor Controller(UAC): 10.1.1.2

User               Port    State                        Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------------
00:00:00:00:00:01  1/1/1   registered                    5          13       10

User Anchor Controller(UAC): 10.1.1.3

User               Port    State                        Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------------
00:00:00:00:00:02  1/1/2   registered                    4          14       20
```

Showing the UBT state of a UAC with IP address `15.212.219.57` where local-VLAN mode has been configured:

```
switch# show ubt state zone zone1 uac-ip 15.212.219.57

User Anchor Controller(UAC): 15.212.219.57

User               Port    State                        Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------------
00:00:00:00:00:04  1/1/20  registered                    4          14       4000
```

Showing the UBT state of a UAC with IP address `15.212.219.55` where VLAN-extend mode has been configured:

```
switch# show ubt state zone zone1 uac-ip 15.212.219.55
```

```
User Anchor Controller(UAC): 15.212.219.55

User               Port    State                          Bucket ID  Gre Key  VLAN
-------------------------------------------------------------------------------
00:00:00:00:00:07  1/1/10  registered                     40          14       20
00:00:00:00:00:08  1/1/12  registered                     28          14       30
```

show ubt statistics

Syntax

```
show ubt statistics
show ubt statistics zone <ZONE-NAME>
show ubt statistics zone <ZONE-NAME> uac-ip <UAC-ADDR>
```

Description

Displays statistics for UBT.

Specifying a zone shows the UBT statistics for that zone.

Specifying a UAC IP address shows the UBT statistics for that UAC.

Command context

Operator (>) or Manager (#)

Parameters

`zone <ZONE-NAME>`

Specifies UBT zone name. Maximum characters: 64.

`uac-ip <UAC-ADDR>`

Specifies the IP address of the user anchor controller for which to view user information. Specify the address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255.

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 UBT statistics where local-VLAN mode has been configured:

```
switch# show ubt statistics
UBT Statistics
======================================================================
Zone zone1:
======================================================================
Control Plane Statistics
  Active   : 10.1.1.1
    Bootstrap Tx   :  10              Bootstrap Rx       : 10
    Nodelist Rx    :  25              Nodelist Ack Rx    : 6
    Bucketmap Rx   :  21              Bucketmap Ack Rx   : 10
    Failover Tx    :  4               Failover Ack Rx    : 3
    Unbootstrap Tx :  7               Unbootstrap Ack Rx : 5
    Heartbeat Tx   :  5               Heartbeat Rx       : 3
  Standby  : 10.1.1.2
    Bootstrap Tx   : 10               Bootstrap Rx       : 10
```

```
    Nodelist Rx    : 25               Nodelist Ack Rx    : 6
    Bucketmap Rx   : 21               Bucketmap Ack Rx   : 12
    Failover Tx    : 4                Failover Ack Rx    : 3
    Unbootstrap Tx : 5                Unbootstrap Ack Rx : 3
    Heartbeat Tx   : 7                Heartbeat Rx       : 4
  UAC : 10.1.1.1
    Bootstrap Tx   : 10               Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 5                Unbootstrap Ack Rx : 5
    Keepalive Tx   : 2                Keepalive Ack Rx   : 2
  UAC : 10.1.1.2
    Bootstrap Tx   : 5                Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 0                Unbootstrap Ack Rx : 0
    Keepalive Tx   : 0                Keepalive Ack Rx   : 0
Data Plane Statistics
  UAC       Packets Tx   Packets Rx
  ---------------------------------
  10.1.1.1  45678        23456
  10.1.1.2  34567        23457
User Statistics
  UAC       User Count
  ----------------------
  10.1.1.1   1
  10.1.1.2   2
=============================================================================
Zone zone2:
=============================================================================
Control Plane Statistics
  Active   : 20.1.1.3
    Bootstrap Tx   :  10              Bootstrap Rx       : 10
    Nodelist Rx    :  25              Nodelist Ack Rx    : 6
    Bucketmap Rx   :  21              Bucketmap Ack Rx   : 10
    Failover Tx    :  4               Failover Ack Rx    : 3
    Unbootstrap Tx :  7               Unbootstrap Ack Rx : 5
    Heartbeat Tx   :  5               Heartbeat Rx       : 3
  Standby  : 20.1.1.4
    Bootstrap Tx   : 10               Bootstrap Rx       : 10
    Nodelist Rx    : 25               Nodelist Ack Rx    : 6
    Bucketmap Rx   : 21               Bucketmap Ack Rx   : 12
    Failover Tx    : 4                Failover Ack Rx    : 3
    Unbootstrap Tx : 5                Unbootstrap Ack Rx : 3
    Heartbeat Tx   : 7                Heartbeat Rx       : 4
  UAC : 20.1.1.3
    Bootstrap Tx   : 10               Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 5                Unbootstrap Ack Rx : 5
    Keepalive Tx   : 2                Keepalive Ack Rx   : 2
  UAC : 20.1.1.4
    Bootstrap Tx   : 5                Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 0                Unbootstrap Ack Rx : 0
    Keepalive Tx   : 0                Keepalive Ack Rx   : 0
Data Plane Statistics
  UAC       Packets Tx   Packets Rx
  ---------------------------------
  20.1.1.3  45670        33456
  20.1.1.4  34561        33457
User Statistics
  UAC       User Count
  ----------------------
```

```
  20.1.1.3  1
  20.1.1.4  2
```

Showing UBT statistics where VLAN-extend mode has been configured:

```
switch# show ubt statistics
UBT Statistics
================================================================================
Zone zone1:
================================================================================
Control Plane Statistics
  Active   : 10.1.1.3
    Bootstrap Tx   :  10              Bootstrap Rx       : 10
    Nodelist Rx    :  25              Nodelist Ack Rx    : 6
    Bucketmap Rx   :  21              Bucketmap Ack Rx   : 10
    Failover Tx    :  4               Failover Ack Rx    : 3
    Unbootstrap Tx :  7               Unbootstrap Ack Rx : 5
    Heartbeat Tx   :  5               Heartbeat Rx       : 3
  Standby  : 10.1.1.4
    Bootstrap Tx   : 10               Bootstrap Rx       : 10
    Nodelist Rx    : 25               Nodelist Ack Rx    : 6
    Bucketmap Rx   : 21               Bucketmap Ack Rx   : 12
    Failover Tx    : 4                Failover Ack Rx    : 3
    Unbootstrap Tx : 5                Unbootstrap Ack Rx : 3
    Heartbeat Tx   : 7                Heartbeat Rx       : 4
  UAC : 10.1.1.3
    Bootstrap Tx   : 10               Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 5                Unbootstrap Ack Rx : 5
    Keepalive Tx   : 2                Keepalive Ack Rx   : 2
  UAC : 10.1.1.4
    Bootstrap Tx   : 5                Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 0                Unbootstrap Ack Rx : 0
    Keepalive Tx   : 0                Keepalive Ack Rx   : 0
Data Plane Statistics
  SAC tunnel Rx                   : 444
  Standby-SAC tunnel Rx           : 0
  UAC        Packets Tx   Packets Rx
  ---------------------------------
  10.1.1.3   45678        23456
  10.1.1.4   34567        23457
User Statistics
  UAC       User Count
  ----------------------
  10.1.1.3   1
  10.1.1.4   2
================================================================================
Zone zone2:
================================================================================
Control Plane Statistics
  Active   : 20.1.1.3
    Bootstrap Tx   :  10              Bootstrap Rx       : 10
    Nodelist Rx    :  25              Nodelist Ack Rx    : 6
    Bucketmap Rx   :  21              Bucketmap Ack Rx   : 10
    Failover Tx    :  4               Failover Ack Rx    : 3
    Unbootstrap Tx :  7               Unbootstrap Ack Rx : 5
    Heartbeat Tx   :  5               Heartbeat Rx       : 3
```

```
  Standby  : 20.1.1.4
    Bootstrap Tx   : 10                Bootstrap Rx       : 10
    Nodelist Rx    : 25                Nodelist Ack Rx    : 6
    Bucketmap Rx   : 21                Bucketmap Ack Rx   : 12
    Failover Tx    : 4                 Failover Ack Rx    : 3
    Unbootstrap Tx : 5                 Unbootstrap Ack Rx : 3
    Heartbeat Tx   : 7                 Heartbeat Rx       : 4
  UAC : 20.1.1.3
    Bootstrap Tx   : 10                Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 5                 Unbootstrap Ack Rx : 5
    Keepalive Tx   : 2                 Keepalive Ack Rx   : 2
  UAC : 20.1.1.4
    Bootstrap Tx   : 5                 Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 0                 Unbootstrap Ack Rx : 0
    Keepalive Tx   : 0                 Keepalive Ack Rx   : 0
Data Plane Statistics
  SAC tunnel Rx                   : 222
  Standby-SAC tunnel Rx           : 0
  UAC       Packets Tx   Packets Rx
  ---------------------------------
  20.1.1.3  45678        23456
  20.1.1.4  34567        23457
User Statistics
  UAC       User Count
  ------------------------
  20.1.1.3   1
  20.1.1.4   2
```

Showing UBT statistics for `zone1` where local-vlan mode has been configured:

```
switch# show ubt statistics zone zone1

UBT Statistics

Zone zone1:
Control Plane Statistics

  Active   : 10.1.1.3
    Bootstrap Tx   :  10               Bootstrap Rx       : 10
    Nodelist Rx    :  25               Nodelist Ack Rx    : 6
    Bucketmap Rx   :  21               Bucketmap Ack Rx   : 10
    Failover Tx    :  4                Failover Ack Rx    : 3
    Unbootstrap Tx :  7                Unbootstrap Ack Rx : 5
    Heartbeat Tx   :  5                Heartbeat Rx       : 3

  Standby  : 10.1.1.4
    Bootstrap Tx   : 10                Bootstrap Rx       : 10
    Nodelist Rx    : 25                Nodelist Ack Rx    : 6
    Bucketmap Rx   : 21                Bucketmap Ack Rx   : 12
    Failover Tx    : 4                 Failover Ack Rx    : 3
    Unbootstrap Tx : 5                 Unbootstrap Ack Rx : 3
    Heartbeat Tx   : 7                 Heartbeat Rx       : 4

  UAC : 10.1.1.3
    Bootstrap Tx   : 10                Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 5                 Unbootstrap Ack Rx : 5
    Keepalive Tx   : 2                 Keepalive Ack Rx   : 2

  UAC : 10.1.1.4
```

```
    Bootstrap Tx   : 5                Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 0                Unbootstrap Ack Rx : 0
    Keepalive Tx   : 0                Keepalive Ack Rx   : 0

Data Plane Statistics

  UAC       Packets Tx   Packets Rx
  ---------------------------------
  10.1.1.3  45678        23456
  10.1.1.4  34567        23457

User Statistics

  UAC       User Count
  ----------------------
  10.1.1.3   1
  10.1.1.4   2
```

Showing UBT statistics for `zone1` where VLAN-extend mode has been configured:

```
switch# show ubt statistics zone zone1

UBT Statistics

Zone zone1:
Control Plane Statistics

  Active   : 10.1.1.3
    Bootstrap Tx   :  10              Bootstrap Rx       : 10
    Nodelist Rx    :  25              Nodelist Ack Rx    : 6
    Bucketmap Rx   :  21              Bucketmap Ack Rx   : 10
    Failover Tx    :  4               Failover Ack Rx    : 3
    Unbootstrap Tx :  7               Unbootstrap Ack Rx : 5
    Heartbeat Tx   :  5               Heartbeat Rx       : 3

  Standby  : 10.1.1.4
    Bootstrap Tx   : 10               Bootstrap Rx       : 10
    Nodelist Rx    : 25               Nodelist Ack Rx    : 6
    Bucketmap Rx   : 21               Bucketmap Ack Rx   : 12
    Failover Tx    : 4                Failover Ack Rx    : 3
    Unbootstrap Tx : 5                Unbootstrap Ack Rx : 3
    Heartbeat Tx   : 7                Heartbeat Rx       : 4

  UAC : 10.1.1.3
    Bootstrap Tx   : 10               Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 5                Unbootstrap Ack Rx : 5
    Keepalive Tx   : 2                Keepalive Ack Rx   : 2

  UAC : 10.1.1.4
    Bootstrap Tx   : 5                Bootstrap Ack Rx   : 5
    Unbootstrap Tx : 0                Unbootstrap Ack Rx : 0
    Keepalive Tx   : 0                Keepalive Ack Rx   : 0

Data Plane Statistics

  SAC tunnel Rx                   : 444
  Standby-SAC tunnel Rx           : 0

  UAC        Packets Tx   Packets Rx
  ---------------------------------
  10.1.1.3   45678        23456
```

```
  10.1.1.4   34567        23457

User Statistics

  UAC       User Count
  ----------------------
  10.1.1.3   1
  10.1.1.4   2
```

Showing the UBT statistics of a UAC with IP address `101.101.101.11`:

```
switch# show ubt statistics zone zone1 uac-ip 101.101.101.11
Data Plane Statistics

  SAC tunnel Rx                  : 6457
  Standby-SAC tunnel Rx          : 0

  UAC               Packets Tx          Packets Rx
  ------------------------------------------------

  101.101.101.11  : 145379605           145450113
```

show ubt users

Syntax

`show ubt users [ all | count | down | mac <MAC-ADDR> | {port <IF-NAME> | <IF-RANGE>} | up] zone <ZONE-NAME>`

Description

Displays user information for UBT.

Command context

Operator (`>`) or Manager (`#`)

Parameters

`all`
Display information for all users.

`count`
Display the total number of users configured to tunnel traffic.

`down`
Display the users that are not able to tunnel traffic.

`mac <MAC-ADDR>`
Display user information based on MAC address.

`port <IF-NAME> | <IF-RANGE>`
Display user information for a specific interface or range of interfaces. For example, `port 1/1/1` or `port 1/1/1-1/1/10`.

`up`
Display user information that are active.

`zone <ZONE-NAME>`
Specifies UBT zone name. Maximum characters: 64.

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

On the 6400 Switch Series, interface identification differs.

Showing information for all users:

```
switch# show ubt users all
================================================================================
Displaying All UBT Users for Zone: zone1
================================================================================
Downloaded user roles are preceded by *
Port    Mac Address        Tunnel Status  Secondary UserRole   Failure Reason
--------------------------------------------------------------------------------
1/25    00:00:00:11:12:03  activated      authenticated         ---/---
================================================================================
Displaying All UBT Users for Zone: zone2
================================================================================
Downloaded user roles are preceded by *
Port    Mac Address        Tunnel Status  Secondary UserRole   Failure Reason
--------------------------------------------------------------------------------
2/25    00:00:00:13:12:03  activated      authenticated         ---/---
```

Showing information for users of `zone1`:

```
switch# show ubt users all zone zone1

================================================================================
Displaying All UBT Users for Zone: zone1
================================================================================
Downloaded user roles are preceded by *
Port    Mac Address        Tunnel Status  Secondary UserRole   Failure Reason
--------------------------------------------------------------------------------
1/25    00:00:00:11:12:03  activated      authenticated         ---/---
```

Displaying the number of users that are tunneling traffic:

```
switch# show ubt users count

Total Number of Users using ubt Zone : zone2 is 1

Total Number of Users using ubt Zone : zone1 is 2
================================================
Total Number of Users in all the zones : 3
================================================
```

Showing users that are down:

```
switch# show ubt users down

================================================================================
Displaying UBT Users of Zone: zone1 having Tunnel Status DOWN
================================================================================
Downloaded user roles are preceded by *
```

```
Port    Mac Address       Tunnel Status     Secondary UserRole  Failure Reason
-------------------------------------------------------------------------------
1/25    00:00:00:11:12:03 activation_failed authenticated       PBF Failure
```

Showing information for users of `zone1` that are down:

```
switch# show ubt users down zone zone1

================================================================================
Displaying UBT Users of Zone: zone1 having Tunnel Status DOWN
================================================================================
Downloaded user roles are preceded by *
Port    Mac Address       Tunnel Status     Secondary UserRole  Failure Reason
-------------------------------------------------------------------------------
1/25    00:00:00:11:12:03 activation_failed authenticated       PBF Failure
```

Showing information for users on port `2/25`:

```
switch# show ubt users port 2/25

================================================================================
Displaying UBT Users of Zone: zone1
================================================================================
Downloaded user roles are preceded by *
Port    Mac Address        Tunnel Status  Secondary UserRole  Failure Reason
-------------------------------------------------------------------------------
2/25    00:00:00:11:12:03   activated      authenticated       ---/---
```

Showing information for users that are up:

```
switch# show ubt users up

================================================================================
Displaying UBT Users of Zone: zone1 having Tunnel Status UP
================================================================================
Downloaded user roles are preceded by *
Port    Mac Address       Tunnel Status  Secondary UserRole  Failure Reason
-------------------------------------------------------------------------------
1/25    00:00:00:11:12:03 activated      authenticated       ---/---
```

Showing information for users of `zone1` that are up:

```
switch# show ubt users up zone zone1

================================================================================
Displaying UBT Users of Zone: zone1 having Tunnel Status UP
================================================================================
Downloaded user roles are preceded by *
Port    Mac Address       Tunnel Status  Secondary UserRole  Failure Reason
-------------------------------------------------------------------------------
1/25    00:00:00:11:12:03 activated      authenticated       ---/---
```

Showing information for the user with MAC address `00:00:00:11:12:03`:

```
switch# show ubt users mac 00:00:00:11:12:03

Displaying UBT User of Zone: zone1 having MAC-Address: 00:00:00:11:12:03
Downloaded user roles are preceded by *
Port    Mac Address       Tunnel Status  Secondary UserRole  Failure Reason
------------------------------------------------------------------------
1/25    00:00:00:11:12:03 activated      authenticated       ---/---
```

uac-keepalive-interval

Syntax

```
uac-keepalive-interval <TIME>
no uac-keepalive-interval <TIME>
```

Description

Specifies the UAC keep alive refresh time interval in seconds for the UBT zone.

The `no` form of this command sets the keep alive interval to the default value.

Command context

`config-ubt-<ZONE-NAME>`

Parameters

<TIME>

Specifies the UAC keep-alive refresh time interval in seconds. Range: 1 to 60. Default: 60.

Authority

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

Examples

Specifying a keepalive interval of 60 seconds for UBT `zone1`:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# uac-keepalive-interval 60
```

Deleting the configured UAC keepalive interval:

```
switch(config)# ubt zone zone1
switch(config-ubt-zone1)# no uac-keepalive-interval 60
```

ubt

Syntax

```
ubt zone <ZONE-NAME> vrf <VRF-NAME>
no ubt zone <ZONE-NAME> vrf <VRF-NAME>
```

Description

Creates a User Based Tunnel (UBT) zone with a specified zone name and VRF name. A UBT name is used to configure all UBT properties advertised by the UBT feature.

The `no` form of this command removes the specified UBT zone.

Command context

`config`

Parameters

<ZONE-NAME>

Specifies a name for the UBT zone. Length: 1 to 64 characters.

<VRF-NAME>

Specifies the VRF on which to establish the UBT tunnel.

Authority

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

Examples

Creating UBT zone called `zone1` associated with a VRF called `default`:

```
switch(config)# ubt zone zone1 vrf default
```

Removing UBT zone `zone1` on VRF `default`:

```
switch(config)# no ubt zone zone1 vrf default
```

Deleting all UBT configurations:

```
switch(config)# no ubt
```

ubt-client-vlan

Syntax

`ubt-client-vlan` *<VLAN-ID>*

`no ubt-client-vlan` *<VLAN-ID>*

Description

Specifies the UBT Client VLAN or local VLAN. This VLAN is used in local-VLAN mode only. If the UBT client VLAN is configured in VLAN-extend mode it is ignored. No other feature should be enabled on the UBT client VLAN.

The `no` form of this command removes the VLAN to use for tunneled clients.

Command context

`config`

Parameters

<VLAN-ID>

Specifies the VLAN ID to use for tunneled clients. Range: 1-4094.

Authority

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

Examples

Creating VLAN 4000:

```
switch(config)# vlan 4000
switch(config-vlan-4000)# no shutdown
```

Specifying UBT client VLAN 4000:

```
switch(config)# ubt-client-vlan 4000
```

Removing configured UBT client VLAN 4000:

```
switch(config)# no ubt-client-vlan 4000
```

ubt mode vlan-extend

Syntax

```
ubt-mode vlan-extend

no ubt-mode
```

Description

Selects VLAN extended mode. When VLAN-extend mode is enabled clients are assigned to their UBT role-based VLAN in the hardware datapath.

The `no` form of the command selects local-VLAN mode. In local-VLAN mode clients are assigned to a local switch VLAN and associated with their UBT role-based VLAN when client traffic reaches the controller.

The default UBT mode is local-VLAN.

Command context

`config`

Authority

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

Examples

Setting the UBT mode to VLAN-extend:

```
switch(config)# ubt-mode vlan-extend
```

Setting the UBT mode back to the default of local-VLAN:

```
switch(config)# no ubt-mode
```

Simple Network Management Protocol (SNMP) is an Internet-standard protocol used for managing and monitoring the devices connected to a network by collecting, organizing and modifying information about managed devices on IP networks.

Configuring SNMP

(The SNMP agent provides read-only access.)

Procedure

1. Enable SNMP on a VRF using the command `snmp-server vrf`.
2. Set the system contact, location, and description for the switch with the following commands:
   - `snmp-server system-contact`
   - `snmp-server system-location`
   - `snmp-server system-description`
3. If required, change the default SNMP port on which the agent listens for requests with the command `snmp-server agent-port`.
4. By default, the agent uses the community string **public** to protect access through SNMPv1/v2c. Set a new community string with the command `snmp-server community`.
5. Configure the trap receivers to which the SNMP agent will send trap notifications with the command `snmp-server host`.
6. Create an SNMPv3 context and associate it with any available SNMPv3 user to perform context specific v3 MIB polling using the command `snmpv3 user .`
7. Create an SNMPv3 context and associate it with an available SNMPv1/v2c community string to perform context specific v1/v2c MIB polling using the command `snmpv3 context`.
8. Review your SNMP configuration settings with the following commands:
   - `show snmp agent-port`
   - `show snmp community`
   - `show snmp system`
   - `show snmpv3 context`
   - `show snmp trap`
   - `show snmp vrf`
   - `show snmpv3 users`
   - `show tech snmp`

Example 1

This example creates the following configuration:

- Enables SNMP on the out-of-band management interface (VRF **mgmt**).
- Sets the contact, location, and description for the switch to: **JaniceM**, **Building2**, **LabSwitch**.
- Sets the community string to **Lab8899X**.

```
switch(config)# snmp-server vrf mgmt
switch(config)# snmp-server system-contact JaniceM
switch(config)# snmp-server system-location Building2
switch(config)# snmp-server system-description LabSwitch
switch(config)# snmp-server community Lab8899X
```

Example 2

This example creates the following configuration:

- Creates an SNMPv3 user named **Admin** using **sha** authentication with the plaintext password **mypassword** and using **des** security with the plaintext password **myprivpass**.
- Associates the SNMPv3 user `Admin` with a context named `newContext`.

```
switch(config)# snmpv3 user Admin auth sha auth-pass plaintext mypassword priv des
     priv-pass plaintext myprivpass
switch(config)# snmpv3 user Admin context newContext
```

Refer to the SNMP Guide for SNMP Commands.

The Aruba Central network management solution, a software-as-a-service subscription in the cloud, provides streamlined management of multiple network devices. AOS-CX switches are able to talk to Aruba Central and utilize cloud-based management functionality. Cloud-based management functionality allows for the deployment of network devices at sites with no or few dedicated IT personnel (branch offices, retail stores, and so forth). AOS-CX switches utilize secure communication protocols to connect to the Aruba Central cloud portal, and can coexist with corporate security standards, such as those mandating the use of firewalls.

When Aruba Central manages AOS-CX switches, it functions as the single source of truth and the Web UI operates in read-only mode.

This feature provides:

- Zero-touch provisioning
- Network Management/Remote monitoring
- Events/alerts notification
- Switch Configuration using templates
- Firmware management

Connecting to Aruba Central

AOS-CX switch downloads the location of Aruba Central server using:

- Command-line interface (CLI).
- Aruba Activate server.
- DHCP options provided during ZTP.

DHCP servers are used to connect to Central on-premise management.

If switch is unable to connect to Activate server, it retries to establish connection in exponential back off of 1s, 2s, 8s, 16s, 32s, 64s, 128s, and 256s. After the maximum back off of 256s, switch retries happen for every 5 minutes.

If the Network Time Protocol (NTP) is not enabled on the switch, it will synchronize the system time with the Activate server.

Custom CA certificate

To use custom CA certificate to connect to Aruba Central, AOS-CX switch downloads the certificate from Aruba Activate server.

- If there is no custom CA provided by Aruba Activate, the CA certificate present in the device is used.
- Duplicate CA certificates from Aruba Activate server will be ignored.
- If CA certificate is absent in consecutive responses from Aruba Activate server, the installed custom CA certificate in device will be removed.
- Switch will have only one custom CA certificate installed from Aruba Activate Server.
- The certificate installed from Aruba Activate server will not be displayed in the show commands.

Support mode in Aruba Central

When the AOS-CX switch is managed by Aruba Central, the switch configuration cannot be modified using other interfaces such as CLI or Web UI. The following command categories are blocked:

- auto-confirm
- boot
- checkpoint
- copy-in commands
- erase
- erps
- https-server
- mfgread
- mfgwrite
- port-access
- vsf
- All configuration commands except the aruba-central command

In cases where a maintenance or troubleshooting activity requires configuration updates, aruba-central support-mode can be enabled to allow these operations.

The aruba-central support-mode enable or disable operation is effective only in the CLI session where it is executed and does not impact the other CLI sessions.

If the user tries to execute any command that is not allowed, an **Invalid input:** error message is displayed.

Aruba Central commands

aruba-central

Syntax

```
aruba-central

no aruba-central
```

Description

Creates or enters the Aruba Central configuration context (`config-aruba-central`).

Command context

```
config
```

Authority

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

Example

Creating the Aruba Central configuration context:

```
switch(config)# aruba-central
switch(config-aruba-central)#
```

aruba-central support-mode

Syntax

```
aruba-central support-mode

no aruba-central support-mode
```

Description

Allows the device to be writable for all operations in Aruba Central lockout mode for troubleshooting. The no form of this command disables this activity.

Support-mode is disabled by default when the switch is managed by Aruba Central. This command is only effective in the CLI session where it is executed.

Command context

Manager (#)

Authority

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

Examples

Configuring the device to be writable for all operations in Aruba Central lockout mode:

```
switch# aruba-central support-mode
switch#
```

Removing the configuration that allows the device to be writable for all operations in Aruba Central lockout mode:

```
switch# no aruba-central support-mode
switch#
```

configuration-lockout central managed

Syntax

```
configuration-lockout central managed
```

```
no configuration-lockout central managed
```

Description

Configures the device to only be writable from Aruba Central. Aruba Central will be the only agent that can add, modify, or delete configurations on the device. The no form of this command disables this feature.

The no form of this command is only available when the device is disconnected from Aruba Central.

Command context

```
config
```

Authority

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

Usage

The AOS-CX switch connects to Aruba Central in either of two modes: monitor or managed. When the device is connected in monitor mode, Aruba Central monitors the configurations on the switch. When the device is connected in managed mode, the `configuration-lockout central managed` command does not allow configuration changes from other interfaces such as CLI or Web UI.

Examples

Configuring the device to only be writable from Aruba Central :

```
switch(config)# configuration-lockout central managed
switch# show configuration-lockout
configuration lockout
---------------------
central: managed
switch# sh aruba-central
Central admin state                 :enable
Central location                    :20.0.0.2:8083
VRF for connection                  :default
Central connection status           :connected

Central source                      :cli
Central source connection status    :connected
Central source last connected on    :Tue Feb 9 17:53:13 UTC 2021

Activate Server URL                 :devices-v2.arubanetworks.com
CLI location                        :20.0.2:8083
CLI VRF                             :default
switch(config)# end
```

disable

Syntax

```
disable
```

Description

Disables connection to Aruba Central server.

When the connection is disabled, the switch does not attempt to connect to the Aruba Central server or fetch central location from any of the three sources (CLI/Aruba Activate/DHCP). It also disconnects any active connection to the Aruba Central server.

Command context

`config-aruba-central`

Authority

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

Example

```
switch(config-aruba-central)# disable
switch(config-aruba-central)#
```

enable

Syntax

`enable`

Description

Enables connection to Aruba Central server. When the connection is enabled, the switch attempts to download the location of the Aruba Central server in one of the following ways at startup and after the connection is lost:

- Using command-line interface (CLI).
- Connecting to Aruba Activate server.
- Using DHCP options provided during ZTP.

DHCP servers provide the options requested by the device to connect to Central, Central On-premise managment, or the TFTP server.

Command context

`config-aruba-central`

Authority

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

Examples

```
switch(config-aruba-central)# enable
switch(config-aruba-central)#
```

location-override

Syntax

```
location-override <location> [vrf <VRF-NAME>]
 no location-override
```

Description

When `location` and `vrf` are configured, the switch overrides existing connections to Aruba Central. The switch attempts to establish connection to Aruba Central with the specified location and VRF with highest priority.

The `no` form of this command removes location override values from the Aruba Central configuration context.

Command context

`config-aruba-central`

Authority

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

Parameters

`<location>`

Specifies one of these values:

- *`<FQDN>`*: a fully qualified domain name.
- *`<IPV4>`*: an IPv4 address.
- *`<IPV6>`*: an IPv6 address.

`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.

Examples

Configuring location override with location and VRF:

```
switch(config-aruba-central)# location-override aruba-central.com vrf default
switch(config-aruba-central)#
```

Configuring location override with location only:

```
switch(config-aruba-central)# location-override aruba-central.com
switch(config-aruba-central)#
```

Removing location override values from the Aruba Central configuration context:

```
switch(config-aruba-central)# no location-override
switch(config-aruba-central)#
```

show aruba-central

Syntax

`show aruba-central`

Description

Shows information about Aruba Central connection and the status of the Activate server connection.

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

Example of a switch that has the Aruba Central connection:

```
switch# show aruba-central
Central admin state                    :enabled
Central location                       : N/A
VRF for connection                     : N/A
Central connection status              : N/A
Central source                         : dhcp
Central source connection status       : connection_failure
Central source last connected on       : N/A
System time synchronized from Activate : True
Activate server URL                    : 172.17.0.1
CLI location                           : N/A
CLI VRF                                : N/A
Source IP                              : N/A
Source IP Overridden                   : false
Central support mode                   : disabled
```

show running-config current-context

Syntax

```
show running-config current-context
```

Description

Shows the running configuration for the current-context. If user is in the context of Aruba-Central(`config-aruba-central`), then Aruba Central running configuration is displayed.

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.

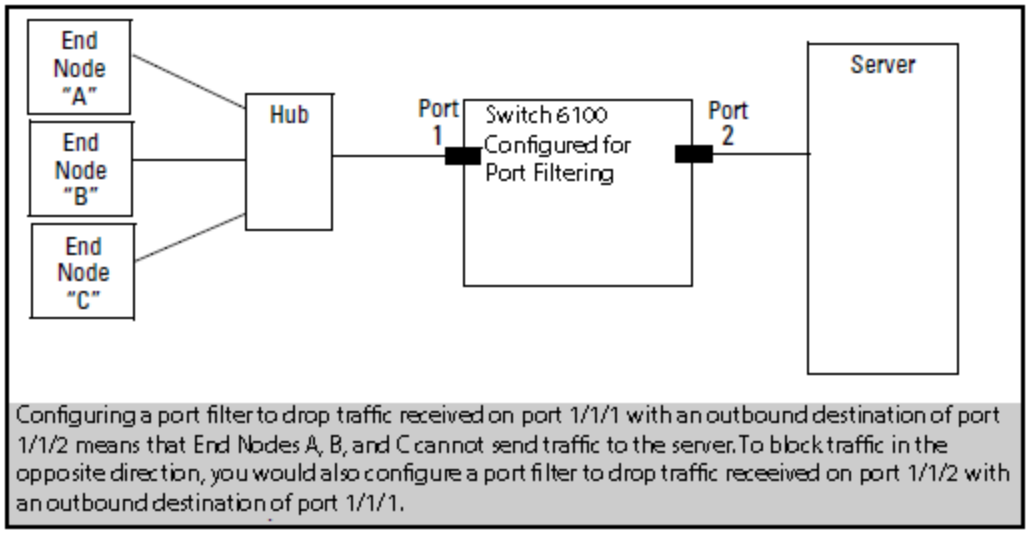
Examples

Shows the running configuration of Aruba Central:

```
switch(config-aruba-central)# show running-config current-context
aruba-central
       disable
```

Port filtering is a feature in which packets that are ingressed through a source port can be blocked for egressing on a specific set of ports.

Figure 1 *Port Filter Application*



;" />

Port filtering commands

portfilter

Syntax

```
portfilter <INTERFACE-LIST>

no portfilter [<INTERFACE-LIST>]
```

Description

Configures the specified ports so they do not egress any packets that were received on the source port specified in interface context.

The `no` form of this command removes the port filter setting from one or more ingress ports/LAGs.

Command context

```
config-if
config-lag-if
```

Parameters

<INTERFACE-LIST>

Specifies a list of ports/LAGs to be blocked for egressing. Specify a single interface or LAG, or a range as a comma-separated list, or both. For example: `1/1/1`, `1/1/3-1/1/6`,`lag2`, `lag1-lag4`.

On the 6400 Switch Series, interface identification differs.

Authority

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

Usage

When a port filter configuration is applied on the same ingress physical port/LAG, the configuration is updated with the new sets of egress ports/LAGs that are to be blocked for egressing and that are not a part of its previous configuration. Duplicate updates on an existing port filter configuration are ignored.

When egress ports/LAGs are removed from the existing port filter configuration of an ingress port/LAG, egressing is allowed again on those egress ports/LAGs for all packets originating from the ingress port/LAG.

The `no portfilter [<IF-NAME-LIST>]` command removes port filter configurations from the egress ports/LAGs listed in the *<IF-NAME-LIST>* parameter only. All other egress ports/LAGs in the port filter configuration of the ingress port/LAG remain intact.

If no physical ports or LAGs are provided for the `no portfilter` command, the command removes the entire port filter configuration for the ingress port/LAG.

Examples

On the 6400 Switch Series, interface identification differs.

Creating a filter that prevents packets received on port **1/1/1** from forwarding to ports **1/1/3-1/1/6** and to LAGs **1** through **4**:

```
switch(config)# interface 1/1/1
switch(config-if)# portfilter 1/1/3-1/1/6,lag1-lag4
```

Creating a filter that prevents packets received on LAG **1** from forwarding to ports **1/1/6** and LAGs **2** and **4**:

```
switch(config)# interface lag 1
switch(config-lag-if)# portfilter 1/1/6,lag2,lag4
```

Removing filters from an existing configuration that allows back packets received on port **1/1/1** to forward to ports **1/1/6** and LAGs **3** and **4**:

```
switch(config)# interface 1/1/1
switch(config-if)# no portfilter 1/1/6,lag3,lag4
```

Removing all filters from an existing configuration that allows back packets received on LAG **1** to forward to all the ports and LAGs:

```
switch(config)# interface lag 1
switch(config-lag-if)# no portfilter
```

show portfilter

Syntax

`show portfilter [<IFNAME>][vsx-peer]`

Description

Displays filter settings for all interfaces or a specific interface.

Command context

Operator (>) or Manager (#)

Parameters

<IFNAME>

Specifies the ingress interface 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

On the 6400 Switch Series, interface identification differs.

Displaying all port filter settings on the switch:

```
switch# show portfilter
Incoming   Blocked
Interface  Outgoing Interfaces
-------------------------------------------------------------------------------
1/1/1      1/1/3-1/1/6,lag1-lag2
1/1/3      1/1/1,1/1/5,1/1/7,1/1/9,1/1/11,1/1/13,1/1/15,1/1/17,1/1/19,1/1/21,
           1/1/23,1/1/25,1/1/27,1/1/29,1/1/31,1/1/33,1/1/35
lag2       1/1/1,1/1/3-1/1/6
```

Displaying the port filter settings for port **1/1/1**:

```
switch# show portfilter 1/1/1
Incoming   Blocked
Interface  Outgoing Interfaces
-------------------------------------------------------------------------------
1/1/1      1/1/3-1/1/6,lag1-lag2
```

Displaying the port filter settings for **LAG2**:

```
switch# show portfilter lag2
Incoming   Blocked
Interface  Outgoing Interfaces
-------------------------------------------------------------------------------
lag2       1/1/1,1/1/3-1/1/6
```

The Domain Name System (DNS) is the Internet protocol for mapping a hostname to its IP address. DNS allows users to enter more readily memorable and intuitive hostnames, rather than IP addresses, to identify devices connected to a network. It also allows a host to keep the same hostname even if it changes its IP address.

Hostname resolution can be either static or dynamic.

- In static resolution, a local table is defined on the switch that associates hostnames with their IP addresses. Static tables can be used to speed up the resolution of frequently queried hosts.
- Dynamic resolution requires that the switch query a DNS server located elsewhere on the network. Dynamic name resolution takes more time than static name resolution, but requires far less configuration and management.

DNS client

The DNS client resolves hostnames to IP addresses for protocols that are running on the switch. When the DNS client receives a request to resolve a hostname, it can do so in one of two ways:

- Forward the request to a DNS name server for resolution.
- Reply to the request without using a DNS name server, by resolving the name using a statically defined table of hostnames and their associated IP addresses.

Configuring the DNS client

Procedure

1. Configure one or more DNS name servers with the command `ip dns server`.
2. To resolve DNS requests by appending a domain name to the requests, either configure a single domain name with the command `ip dns domain-name`, or configure a list of up to six domain names with the command `ip dns domain-list`.
3. To use static name resolution for certain hosts, associate an IP address to a host with the command `ip dns host`.
4. Review your DNS configuration settings with the command `show ip dns`.

Examples

This example creates the following configuration:

- Defines the domain **switch.com** to append to all requests.
- Defines a DNS server with IPv4 address of **1.1.1.1**.
- Defines a static DNS host named **myhost1** with an IPv4 address of **3.3.3.3**.
- DNS client traffic is sent on the default VRF (named **default**).

```
switch(config)# ip dns domain-name switch.com
switch(config)# ip dns server-address 1.1.1.1
switch(config)# ip dns host myhost1 3.3.3.3
switch(config)# exit
switch# show ip dns

VRF Name : vrf_mgmt

Host Name                                                                    Address
-------------------------------------------------------------------------------------


VRF Name : vrf_default
Domain Name : switch.com
DNS Domain list :
Name Server(s) : 1.1.1.1

Host Name                                                                    Address
-------------------------------------------------------------------------------------
myhost1
```

This example creates the following configuration:

- Defines three domains to append to DNS requests **domain1.com**, **domain2.com**, **domain3.com** with traffic forwarding on VRF **mainvrf**.
- Defines a DNS server with an IPv6 address of **c::13**.
- Defines a DNS host named **myhost** with an IPv4 address of **3.3.3.3**.

```
switch(config)# ip dns domain-list domain1.com vrf mainvrf
switch(config)# ip dns domain-list domain2.com vrf mainvrf
switch(config)# ip dns domain-list domain3.com vrf mainvrf
switch(config)# ip dns server-address c::13
switch(config)# ip dns host myhost 3.3.3.3 vrf mainvrf
switch(config)# quit
switch# show ip dns mainvrf

VRF Name : mainvrf
Domain Name :
DNS Domain list : domain1.com, domain2.com, domain3.com
Name Server(s) : c::13

Host Name                                                                    Address
-------------------------------------------------------------------------------------
myhost                                                                       3.3.3.3
```

DNS client commands

ip dns domain-list

Syntax

```
ip dns domain-list <DOMAIN-NAME> [vrf <VRF-NAME>]
no ip dns domain-list <DOMAIN-NAME> [vrf <VRF-NAME>]
```

Description

Configures one or more domain names that are appended to the DNS request. The DNS client appends each name in succession until the DNS server replies. Domains can be either IPv4 or IPv6. By default, requests are forwarded on the default VRF.

The `no` form of this command removes a domain from the list.

Command context

`config`

Parameters

`list <DOMAIN-NAME>`

Specifies a domain name. Up to six domains can be added to the list. Length: 1 to 256 characters.

`vrf <VRF-NAME>`

Specifies a VRF name. Default: default.

Authority

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

Examples

This example defines a list with two entries: **domain1.com** and **domain2.com**.

```
switch(config)# ip dns domain-list domain1.com
switch(config)# ip dns domain-list domain2.com
```

This example defines a list with two entries, **domain2.com** and **domain5.com**, with requests being sent on **mainvrf**.

```
switch(config)# ip dns domain-list domain2.com vrf mainvrf
switch(config)# ip dns domain-list domain5.com vrf mainvrf
```

This example removes the entry **domain1.com**.

```
switch(config)# no ip dns domain-list domain1.com
```

ip dns domain-name

Syntax

```
ip dns domain-name <DOMAIN-NAME> [ vrf <VRF-NAME> ]
no ip dns domain-name <DOMAIN-NAME> [ vrf <VRF-NAME> ]
```

Description

Configures a domain name that is appended to the DNS request. The domain can be either IPv4 or IPv6. By default, requests are forwarded on the default VRF. If a domain list is defined with the command `ip dns domain-list`, the domain name defined with this command is ignored.

The `no` form of this command removes the domain name.

Command context

`config`

Parameters

<DOMAIN-NAME>

Specifies the domain name to append to DNS requests. Length: 1 to 256 characters.

vrf *<VRF-NAME>*

Specifies a VRF name. Default: default.

Authority

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

Examples

Setting the default domain name to domain.com:

```
switch(config)# ip dns domain-name domain.com
```

Removing the default domain name domain.com:

```
switch(config)# no ip dns domain-name domain.com
```

ip dns host

Syntax

ip dns host *<HOST-NAME>* *<IP-ADDR>* [vrf *<VRF-NAME>*]

no ip dns host *<HOST-NAME>* *<IP-ADDR>* [vrf *<VRF-NAME>*]

Description

Associates a static IP address with a hostname. The DNS client returns this IP address instead of querying a DNS server for an IP address for the hostname. Up to six hosts can be defined. If no VRF is defined, the default VRF is used.

The no form of this command removes a static IP address associated with a hostname.

Command context

config

Parameters

host *<HOST-NAME>*

Specifies the name of a host. Length: 1 to 256 characters.

<IP-ADDR>

Specifies an IP address in IPv4 format (x.x.x.x), where x is a decimal number from 0 to 255, or IPv6 format (xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx), where x is a hexadecimal number from 0 to F.

vrf *<VRF-NAME>*

Specifies a VRF name. Default: default.

Authority

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

Examples

This example defines an IPv4 address of **3.3.3.3** for **host1**.

```
switch(config)# ip dns host host1 3.3.3.3
```

This example defines an IPv6 address of **b::5** for **host 1**.

```
switch(config)# ip dns host host1 b::5
```

This example defines removes the entry for **host 1** with address **b::5**.

```
switch(config)# no ip dns host host1 b::5
```

ip dns server address

Syntax

```
ip dns server-address <IP-ADDR> [ vrf <VRF-NAME> ]

no ip dns server-address <IP-ADDR> [ vrf <VRF-NAME> ]
```

Description

Configures the DNS name servers that the DNS client queries to resolve DNS queries. Up to six name servers can be defined. The DNS client queries the servers in the order that they are defined. If no VRF is defined, the default VRF is used.

The `no` form of this command removes a name server from the list.

Command context

```
config
```

Parameters

<IP-ADDR>

Specifies an IP address in IPv4 format (`x.x.x.x`), where `x` is a decimal number from 0 to 255, or IPv6 format (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F.

`vrf` *<VRF-NAME>*

Specifies a VRF name. Default: default.

Authority

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

Examples

This example defines a name server at **1.1.1.1**.

```
switch(config)# ip dns server-address 1.1.1.1
```

This example defines a name server at **a::1**.

```
switch(config)# ip dns server-address a::1
```

This example removes a name server at **a::1**.

```
switch(config)# no ip dns server-address a::1
```

show ip dns

Syntax

`show ip dns [vrf <VRF-NAME>][vsx-peer]`

Description

Shows all DNS client configuration settings or the settings for a specific VRF.

Command context

Manager (#)

Parameters

`vrf <VRF-NAME>`

Specifies the VRF for which to show information. If no VRF is defined, the default VRF 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

These examples define DNS settings and then show how they are displayed with the `show ip dns` command.

```
switch(config)# ip dns domain-name domain.com
switch(config)# ip dns domain-list domain5.com
switch(config)# ip dns domain-list domain8.com
switch(config)# ip dns server-address 4.4.4.4
switch(config)# ip dns server-address 6.6.6.6
switch(config)# ip dns host host3 5.5.5.5
switch(config)# ip dns host host2 2.2.2.2
switch(config)# ip dns host host3 c::12
switch(config)# ip dns domain-name reddomain.com vrf red
switch(config)# ip dns domain-list reddomain5.com vrf red
switch(config)# ip dns domain-list reddomain8.com vrf red
switch(config)# ip dns server-address 4.4.4.5 vrf red
switch(config)# ip dns server-address 6.6.6.7 vrf red
switch(config)# ip dns host host3 5.5.5.6 vrf red
switch(config)# ip dns host host2 2.2.2.3 vrf red
switch(config)# ip dns host host3 c::13 vrf red
switch# show ip dns
VRF Name : default

Domain Name : domain.com
DNS Domain list : domain5.com, domain8.com
Name Server(s) : 4.4.4.4, 6.6.6.6

Host Name    Address
```

```
-------------------------------
host2            2.2.2.2
host3            5.5.5.5
host3            c::12

VRF Name : red

Domain Name : reddomain.com
DNS Domain list : reddomain5.com, reddomain8.com
Name Server(s) : 4.4.4.5, 6.6.6.7

Host Name    Address
-------------------------------
host2            2.2.2.3
host3            5.5.5.6
host3            c::13
```

```
switch(config)# ip dns domain-name domain.com vrf red
switch(config)# ip dns domain-list domain5.com vrf red
switch(config)# ip dns domain-list domain8.com vrf red
switch(config)# ip dns server-address 4.4.4.4 vrf red
switch(config)# ip dns server-address 6.6.6.6 vrf red
switch(config)# ip dns host host3 5.5.5.5 vrf red
switch(config)# no ip dns host host2 2.2.2.2 vrf red
switch(config)# ip dns host host3 c::12 vrf red

switch# show ip dns vrf red
VRF Name : red

Domain Name : domain.com
DNS Domain list : domain5.com, domain8.com
Name Server(s) : 4.4.4.4, 6.6.6.6

Host Name    Address
-------------------------------
host3            5.5.5.5
host3            c::12
```

The switch provides support for LLDP, CDP, and local MAC match by using device profiles to enable automatic discovery and configuration of other devices on the network.

Based on the type of devices connected to the interface, device profiles enable predefined configurations that can be applied to the interface. Connected devices are identified using corresponding protocol packets. When the protocol information on the interface ages, the profile or role is revoked from the interface. Only devices connected directly to the switch are detected and processed to apply a device profile. When a device of a configured type is connected to an interface, the switch automatically applies the corresponding Device profiles .

Local MAC match enables dynamic assignment of client attributes, such QoS and VLANs by using a locally configured authentication repository. Local MAC match involves creating MAC groups that are used to classify connected devices based on MAC address, MAC address mask, and MAC OUI. Local MAC match feature is useful when you do not want to afford RADIUS infrastructure or when you want to use local authentication as a backup method in case the RADIUS server is unreachable.

The following parameters can be configured for each role:

- `associate`: Used to associate captive-portal-profile or policy with the role.
- `auth-mode`: Used to configure authentication mode for the role.

  There is no need to configure `auth-mode` for a plain device profile.

- `mtu`: Used to configure MTU for the role.
- `poe-priority`: Used to configure the PoE priority for the role.
- `trust-mode`: Used to configure trust mode for the role.
- `vlan`: Used to configure VLAN mode for the role.
- `stp-admin-edge-port`: Used to configure STP administrative edge port for the role.

For information on role configurations, see the *Security Guide*.

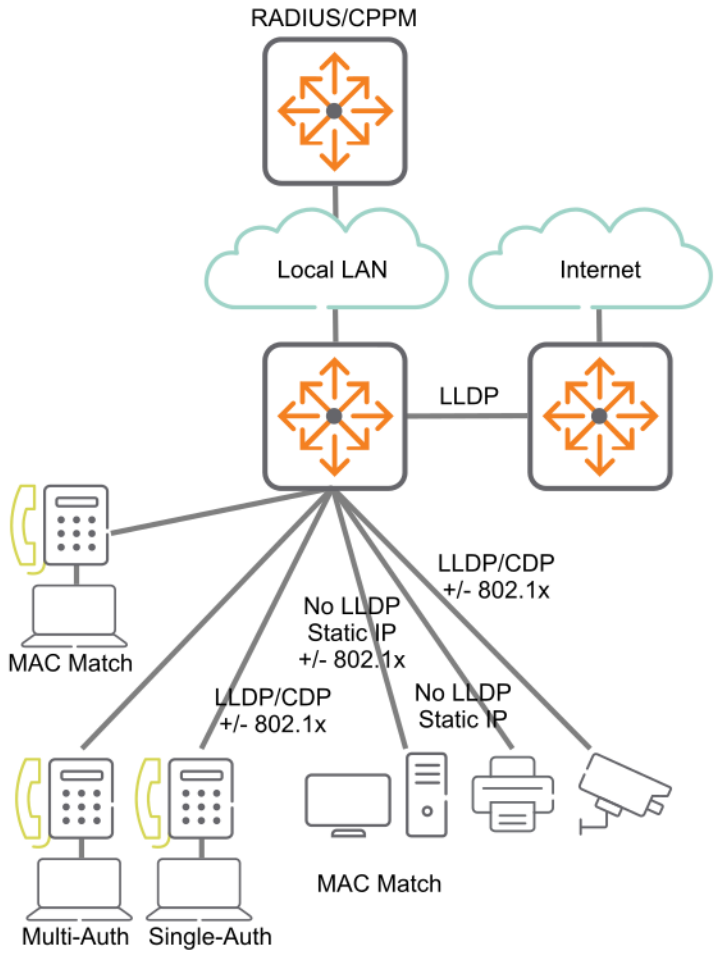
The following commands are not supported in local MAC match feature:

- `aaa authentication port-access mac-auth cached-reauth`
- `aaa authentication port-access mac-auth cached-reauth-period`
- `aaa authentication port-access mac-auth quiet-period`
- `aaa authentication port-access mac-auth reauth-period`

Extra line to keep the above line from following the figure onto the next page.

Example configuration of device deployment

The switch provides simplified deployment of devices, such as access points, IP phones, security cameras, and printers, through the use of a locally configured repository that provides authentication and dynamic port assignment, such as QoS, PoE, and tagged VLANs.



Device profiles

Device profiles rely on role configurations. For information on role configurations, see the *Security Guide*.

Device profiles are used to dynamically assign port attributes based on the type of devices connected, without having to create a RADIUS infrastructure. You can map device profiles to device groups. A device group contains various match criteria, which can be obtained from multiple sources, such as LLDP, CDP, and local MAC match. Device profiles contain port attributes to be assigned to the port when a connected device matches a device group.

Device profiles are supported on different scenarios. It can be applied on interfaces that are configured with security (802.1X or MAC authentication), or applied based on L2 port (LLDP, CDP), or applied on standalone ports with the block-until-profile-applied command enabled. All the methods are mutually exclusive of each other. The block-until-profile-applied mode must be configured only when there is a standalone port where no security has been configured and when you want the port to be offline until at least one client is onboarded based on the match and ignore criteria that you configure. Local MAC match is supported when you configure block-until-profile-applied command or device profile with security.

Up to 16 device profiles can be configured on the 6400.

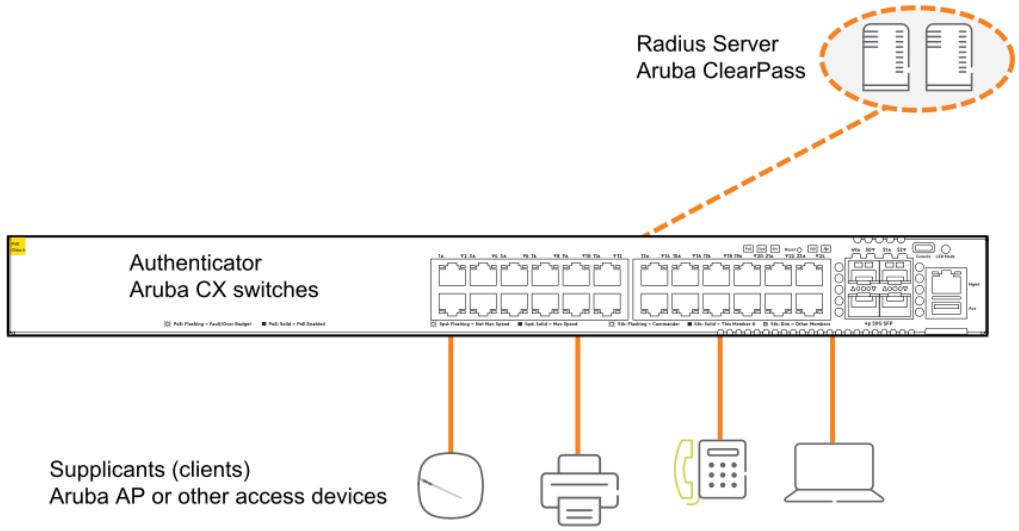
See the *Security Guide* for the following commands:

- The `port-access onboarding-method precedence` command—If you are configuring both security and device profile on the port, and you want to configure the order in which the methods will be executed.
- The `port-access fallback-role` command—If you want to configure a role that must be applied to devices when no other role exists or can be derived for that device.

If you configure a match criteria that matches across multiple device profiles, then the priority considered is LLDP, CDP, and then local MAC match. That is, LLDP precedes over CDP, which in turn precedes over local MAC match.

The following figure displays a simple configuration of device profile and AAA authentication with RADIUS server and Aruba ClearPass Policy Manager. Local MAC match feature is useful when you do not want to afford RADIUS infrastructure or when you want to use local authentication as a backup method in case the RADIUS server is unreachable.

Figure 2 *Example of device profile setup along with RADIUS infrastructure*



;" />

Configuring a device profile for LLDP

Procedure

1. Create an LLDP group with the command `port-access lldp-group`.
2. Define rules for adding devices to an LLDP group with the command `match`.
3. Define rules for ignoring devices so that they are not added to an LLDP group with the command `ignore`.
4. Create a device profile with the command `port-access device-profile`.
5. Add the LLDP group with the command `associate lldp-group`.
6. Add a role to a device profile with the command `associate role`. Make sure that the role is already created. For information on how to create a role, see port access role information in the *Security Guide*.
7. Enable the device profile with the command `enable`.

Configuring a device profile for CDP

1. Create a CDP group with the command `port-access cdp-group`.
2. Define rules for adding devices to a CDP group with the command `match`.

3. Define rules for ignoring devices so that they are not added to a CDP group with the command `ignore`.
4. Create a device profile with the command `port-access device-profile`.
5. Add a CDP group to a device profile with the command `associate cdp-group`.
6. Add a role to a device profile with the command `associate role`. Make sure that the role is already created. For information on how to create a role, see port access role information in the *Security Guide*.
7. Enable a device profile with the command `enable`.

Configuring a device profile for local MAC match

Procedure

1. Create a MAC group with the `mac-group` command.
2. Define rules for adding devices to a MAC group with the `match (for MAC groups)` command.
3. Define rules for ignoring devices so that they are not added to a MAC group with the `ignore (for MAC groups)` command.
4. Create a device profile with the `port-access device-profile` command.
5. Associate a MAC group with a device profile with the `associate mac-group` command.
6. Add a role to a device profile with the `associate role` command. Make sure that the role is already created. For information on how to create a role, see port access role information in the *Security Guide*.
7. Enable a device profile with the `enable` command.

Device profile commands

aaa authentication port-access allow-cdp-bpdu

Syntax

```
aaa authentication port-access allow-cdp-bpdu

no aaa authentication port-access allow-cdp-bpdu
```

Description

Allows all packets related to the CDP (Cisco Discovery Protocol) BPDU (Bridge Protocol Data Unit) on a secure port.

The `no` form of this command blocks the CDP BPDU on a secure port. On a nonsecure port, the command has no effect.

Command context

`config-if`

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Allowing a CDP BPDU on secure port **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# aaa authentication port-access allow-cdp-bpdu
switch(config-if)# do show running-config
Current configuration:
!
!Version ArubaOS-CX 10.0X.0000
led locator on
!
!
vlan 1
aaa authentication port-access mac-auth
    enable
aaa authentication port-access dot1x authenticator
    enable
interface 1/1/1
    no shutdown
    vlan access 1
    aaa authentication port-access allow-cdp-bpdu
    aaa authentication port-access mac-auth
        enable
    aaa authentication port-access dot1x authenticator
    enable

switch(config-if)# do show port-access device-profile interface all
Port 1/1/1, Neighbor-Mac 00:0c:29:9e:d1:20
    Profile Name    : access_switches
    LLDP Group      :
    CDP Group       : aruba-ap_cdp
    Role            : test_ap_role
    Status          : In Progress
    Failure Reason  :
```

Blocking LLDP packet on secure port **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# no aaa authentication port-access allow-cdp-bpdu
switch(config-if)# do show running-config
Current configuration:
!
!Version ArubaOS-CX 10.0X.0000
led locator on
!
!
vlan 1
aaa authentication port-access mac-auth
    enable
interface 1/1/1
    no shutdown
    vlan access 1
    aaa authentication port-access mac-auth
        enable
```

aaa authentication port-access allow-lldp-bpdu

Syntax

```
aaa authentication port-access allow-lldp-bpdu

no aaa authentication port-access allow-lldp-bpdu
```

Description

Allows all packets related to the LLDP BPDU (Bridge Protocol Data Unit) on a secure port.

The `no` form of this command blocks the LLDP BPDU on a secure port. On a nonsecure port, the command has no effect.

Command context

`config-if`

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Allowing an LLDP BPDU on secure port **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# aaa authentication port-access allow-lldp-bpdu
switch(config-if)# do show running-config
Current configuration:
!
!Version ArubaOS-CX 10.0X.0000
led locator on
!
!
vlan 1
aaa authentication port-access mac-auth
    enable
interface 1/1/1
    no shutdown
   vlan access 1
    aaa authentication port-access allow-lldp-bpdu
    aaa authentication port-access mac-auth
        enable

switch(config-if)# do show port-access device-profile interface all
Port 1/1/1, Neighbor-Mac 00:0c:29:9e:d1:20
    Profile Name    : access_switches
    LLDP Group      : 2920-grp
    CDP Group       :
    Role            : local_2920_role
    Status          : Profile Applied
    Failure Reason  :
```

Blocking LLDP BPDU on secure port **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# no aaa authentication port-access allow-lldp-bpdu
switch(config-if)# do show running-config
Current configuration:
!
!Version ArubaOS-CX 10.0X.0000led locator on
!
!
vlan 1
aaa authentication port-access mac-auth
    enable
interface 1/1/1
    no shutdown
```

```
    vlan access 1
    aaa authentication port-access mac-auth
    enable
```

associate cdp-group

Syntax

```
associate cdp-group <GROUP-NAME>

no associate cdp-group <GROUP-NAME>
```

Description

Associates a CDP (Cisco Discovery Protocol) group with a device profile. A maximum of two CDP groups can be associated with a device profile.

The `no` form of this command removes a CDP group from a device profile.

Command context

```
config-device-profile
```

Parameters

<GROUP-NAME>

Specifies the name of the CDP group to associate with this device profile. Range: 1 to 32 alphanumeric characters.

Authority

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

Examples

Associating the CDP group **my-cdp-group** with the device profile **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# associate cdp-group my-cdp-group
```

Removing the CDP group **my-cdp-group** from the device profile **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# no associate cdp-group my-cdp-group
```

associate lldp-group

Syntax

```
associate lldp-group <GROUP-NAME>

no associate lldp-group <GROUP-NAME>
```

Description

Associates an LLDP group with a device profile. A maximum of two LLDP groups can be associated with a device profile

The `no` form of this command removes an LLDP group from a device profile.

Command context

`config-device-profile`

Parameters

<GROUP-NAME>

Specifies the name of the LLDP group to associate with the device profile. Range: 1 to 32 alphanumeric characters.

Authority

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

Examples

Associating the LLDP group **my-lldp-group** with the device profile **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# associate lldp-group my-lldp-group
```

Removing the LLDP group **my-lldp-group** from the device profile **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# no associate lldp-group my-lldp-group
```

associate mac-group

Syntax

`associate mac-group <GROUP-NAME>`

`no associate mac-group <GROUP-NAME>`

Description

Associates a MAC group with a device profile. A maximum of two MAC groups can be associated with a device profile.

The `no` form of this command removes a MAC group from a device profile.

Command context

`config-device-profile`

Parameters

<GROUP-NAME>

Specifies the name of the MAC group to associate with this device profile. Range: 1 to 32 alphanumeric characters.

Authority

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

Examples

Associating the MAC group **mac01-group** with the device profile **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# associate mac-group mac01-group
```

Removing the MAC group **mac01-group** from the device profile **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# no associate mac-group mac01-group
```

associate role

Syntax

```
associate role <ROLE-NAME>

no associate role <ROLE-NAME>
```

Description

Associates a role with a device profile. Only one role can be associated with a device profile. For information on how to configure a role, see the port access role information in the *Security Guide*.

The `no` form of this command removes a role from a device profile.

Command context

```
config-device-profile
```

Parameters

<ROLE-NAME>

Specifies the name of the role to associate with the device profile. Range: 1 to 64 alphanumeric characters.

Authority

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

Examples

Associating the role **my-role** with the device profile **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# associate role my-role
```

Removing the role **my-role** from the device profile **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# no associate role my-role
```

disable

Syntax

```
disable

no disable
```

Description

Disables a device profile.

The `no` form of this command enables a device profile.

Command context

`config-device-profile`

Authority

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

Examples

Disabling a device profile:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# disable
```

Enabling a device profile named profile01:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# no disable
```

enable

Syntax

`enable`

`no enable`

Description

Enables a device profile.

The `no` form of this command disables a device profile.

Command context

`config-device-profile`

Authority

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

Examples

Enabling a device profile:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# enable
```

Disabling a device profile named profile01:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)# no enable
```

ignore (for CDP groups)

Syntax

```
ignore [seq <SEQ-NUM>] {platform <PLATFORM> | sw-version <SWVERSION> |
     voice-vlan-query <VLAN-ID>}

no ignore [seq <SEQ-ID>] {platform <PLATFORM> | sw-version <SWVERSION> |
     voice-vlan-query <VLAN-ID>}
```

Description

Defines a rule to ignore devices for a CDP (Cisco Discovery Protocol) group. Up to 64 match/ignore rules can be defined for a group.

The `no` form of this command removes a rule for ignoring devices from a CDP group.

Command context

```
config-cdp-group
```

Parameters

`seq <SEQ-ID>`

Specifies the ID of the rule to create or modify. If no ID is specified when adding a rule, an ID is automatically assigned in increments of 10 in the order in which rules are added. When more than one rule matches the command entered, the rule with the lowest ID takes precedence.

`platform <PLATFORM>`

Specifies the hardware or model details of the neighbor. Range: 1 to 128 alphanumeric characters.

`sw-version <SWVERSION>`

Specifies the software version of the neighbor. Range: 1 to 128 alphanumeric characters.

`voice-vlan-query <VLAN-ID>`

Specifies the VLAN query value of the neighbor. Range: 1 to 65535.

Authority

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

Examples

Adding a rule to the CDP group **grp01** that ignores a device that transmits **PLATFORM01** in the platform TLV:

```
switch(config)# port-access cdp-group grp01
switch(config-cdp-group)# ignore platform PLATFORM01
```

Adding a rule to the CDP group **grp01** that ignores a device that transmits **SWVERSION** in software version TLV:

```
switch(config)# port-access cdp-group grp01
switch(config-cdp-group)# ignore sw-version SWVERSION
```

Removing the rule that matches the sequence number **25** from the CDP group named **grp01**.

```
switch(config)# port-access cdp-group grp01
switch(config-cdp-group)# no ignore seq 25
```

ignore (for LLDP groups)

Syntax

```
ignore [seq <SEQ-ID>] {sys-desc <SYS-DESC> | sysname <SYS-NAME> |
     vendor-oui <VENDOR-OUI> [type <KEY> [value <VALUE>]]}

no ignore [seq <SEQ-ID>] {sys-desc <SYS-DESC> | sysname <SYS-NAME> |
     vendor-oui <VENDOR-OUI> [type <KEY> [value <VALUE>]]}
```

Description

Defines a rule to ignore devices for an LLDP group. Up to 64 match/ignore rules can be defined for a group.

The `no` form of this command removes a rule for ignoring devices from an LLDP group.

Command context

`config-lldp-group`

Parameters

`seq <SEQ-ID>`

Specifies the ID of the rule to create or modify. If no ID is specified when adding a rule, an ID is automatically assigned in increments of 10 in the order in which rules are added. When more than one rule matches the command entered, the rule with the lowest ID takes precedence.

`sys-desc <SYS-DESC>`

Specifies the LLDP system description type-length-value (TLV). Range: 1 to 256 alphanumeric characters.

`sysname <SYS-NAME>`

Specifies the LLDP system name TLV. Range: 1 to 64 alphanumeric characters.

`vendor-oui <VENDOR-OUI>`

Specifies the LLDP system vendor OUI TLV. Range: 1 to 6 alphanumeric characters.

`type <KEY>`

Specifies the vendor OUI subtype key. Optional.

`value <VALUE>`

Specifies the vendor OUI subtype value. Range: 1 to 256 alphanumeric characters.

Authority

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

Examples

Adding a rule to the LLDP group **grp01** that ignores a device that transmits **PLATFORM01** in the system description TLV:

```
switch(config)# port-access lldp-group grp01
switch(config-lldp-group)# ignore sys-desc PLATFORM01
```

Removing the rule that matches the sequence number **25** from the LLDP group named **grp01**.

```
switch(config)# port-access lldp-group grp01
switch(config-lldp-group)# no match seq 25
```

ignore (for MAC groups)

Syntax

```
[seq <SEQ-ID>] ignore {mac <MAC-ADDR> | mac-mask <MAC-MASK> | mac-oui <MAC-OUI>}

no [seq <SEQ-ID>] ignore {mac <MAC-ADDR> | mac-mask <MAC-MASK> | mac-oui <MAC-OUI>}
```

Description

Defines a rule to ignore devices for a MAC group based on the criteria of MAC address, MAC address mask, or MAC Organizational Unique Identifier (OUI). Up to 64 ignore rules can be defined for a group.

The `no` form of this command removes a rule for ignoring devices from a MAC group.

Command context

`config-mac-group`

Parameters

`seq <SEQ-ID>`

Specifies the entry sequence ID of the rule to create or modify a MAC group. If no ID is specified when adding a rule, an ID is automatically assigned in increments of 10 in the order in which rules are added. When more than one rule matches the command entered, the rule with the lowest ID takes precedence. Range: `1` to `4294967295`.

`mac <MAC-ADDR>`

Specifies the MAC address of the device to ignore.

`mac-mask <MAC-MASK>`

Specifies the MAC address mask to ignore devices in that range. Supported MAC address masks: `/32` and `/40`.

`mac-oui <MAC-OUI>`

Specifies the MAC OUI to ignore devices in that range. Supports MAC OUI address of maximum length of 24 bits.

Authority

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

Usage

To achieve the required configuration of matches for devices, it is recommended to first ignore the devices that you do not want to add. Then match the criteria for the rest of the devices that you want to add to the MAC group.

For example, if you want to ignore a specific device but add all the other devices that belong to a MAC OUI, then you must first configure the ignore criteria with a lower sequence number. And then configure match criteria with a higher sequence number.

Examples

Adding a rule to the MAC group **grp01** to ignore a device based on MAC address, but match all other devices belonging to a MAC OUI:

```
switch(config)# mac-group grp01
switch(config-mac-group)# ignore mac 1a:2b:3c:4d:5e:6f
switch(config-mac-group)# match mac-oui 1a:2b:3c
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
```

```
led locator on
!
!
!
!
ssh server vrf mgmtdefault
!
!
!
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01
     seq 10 ignore mac 1a:2b:3c:4d:5e:6f
     seq 20 match mac-oui 1a:2b:3c

```

Adding a rule to the MAC group **grp01** to ignore devices based on MAC address mask, but match all other devices belonging to a MAC OUI:

```
switch(config)# mac-group grp01
switch(config-mac-group)# ignore mac-mask 1a:2b:3c:4d/32
switch(config-mac-group)# match mac-oui 1a:2b:3c
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
!
!
ssh server vrf mgmtdefault
!
!
!
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01
     seq 10 ignore mac-mask 1a:2b:3c:4d/32
     seq 20 match mac-oui 1a:2b:3c

```

Adding a rule to the MAC group **grp01** that ignores a device based on complete MAC address:

```
switch(config)# mac-group grp01
switch(config-mac-group)# ignore mac 1a:2b:3c:4d:5e:6f
```

Adding a rule to the MAC group **grp02** that ignores devices based on MAC mask:

```
switch(config)# mac-group grp01
switch(config-mac-group)# ignore mac-mask 1a:2b:3c:4d:5e/40
switch(config-mac-group)# ignore mac-mask 18:e3:ab:73/32
```

Adding a rule to the MAC group **grp03** that ignores devices based on MAC OUI:

```
switch(config)# mac-group grp03
switch(config-mac-group)# ignore mac-oui 81:cd:93
```

Adding a rule to the MAC group **grp01** that ignores devices with a sequence number and based on MAC address:

```
switch(config)# mac-group grp01
switch(config-mac-group)# seq 10 ignore mac b2:c3:44:12:78:11
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01
    seq 10 ignore mac b2:c3:44:12:78:11

```
```

Removing the rule from the MAC group **grp01** based on sequence number:

```
switch(config)# mac-group grp01
switch(config-mac-group)# no ignore seq 10
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01

```
```

Adding a rule to the MAC group **grp01** that ignores devices with MAC entry sequence number and based on MAC OUI:

```
switch(config)# mac-group grp01
switch(config-mac-group)# seq 10 ignore mac b2:c3:44:12:78:11
switch(config-mac-group)# seq 20 ignore mac-oui 1a:2b:3c
switch(config-mac-group)# seq 30 ignore mac-mask 71:14:89:f3/32
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01
     seq 10 ignore mac b2:c3:44:12:78:11
     seq 20 ignore mac-oui 1a:2b:3c
     seq 30 ignore mac-mask 71:14:89:f3/32

```

Removing the rule from the MAC group **grp01** based on sequence number and MAC OUI:

```
switch(config)# mac-group grp01
switch(config-mac-group)# no seq 20 ignore mac-oui 1a:2b:3c
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01
     seq 10 ignore mac b2:c3:44:12:78:11
     seq 30 ignore mac-mask 71:14:89:f3/32


```

Removing the rule that matches the sequence number **25** from the MAC group named **grp01**.

```
switch(config)# mac-group grp01
switch(config-mac-group)# no ignore seq 25
```

mac-group

Syntax

`mac-group <MAC-GROUP-NAME>`

`no mac-group <MAC-GROUP-NAME>`

Description

Creates a MAC group or modifies an existing MAC group. A MAC group is used to classify connected devices based on the MAC address details, such as mask or OUI.

A maximum of 32 MAC groups can be configured on the switch. A maximum of 2 MAC groups can be associated with a device profile. Each group accepts 64 match or ignore commands.

The `no` form of this command removes a MAC group.

Command context

`config`

Parameters

`<MAC-GROUP-NAME>`

Specifies the name of the MAC group to create or modify. The maximum number of characters supported is 32.

Authority

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

Examples

Creating a MAC group named `grp01`:

```
switch(config)# mac-group grp01
switch(config-mac-group)# exit
```

Removing a MAC group named `grp01`:

```
switch(config)# no mac-group grp01
```

match (for CDP groups)

Syntax

```
match [seq <SEQ-ID>] {platform <PLATFORM> | sw-version <SWVERSION> |
     voice-vlan-query <VLAN-ID>}

no match [seq <SEQ-ID>] {platform <PLATFORM> | sw-version <SWVERSION> |
     voice-vlan-query <VLAN-ID>}
```

Description

Defines a rule to match devices for a CDP group. A maximum of 32 CDP groups can be configured on the switch. Up to 64 match or ignore rules can be defined for each group.

The `no` form of this command removes a rule for adding devices to a CDP group.

Command context

`config-cdp-group`

Parameters

`seq <SEQ-ID>`

Specifies the ID of the rule to create or modify. If no ID is specified when adding a rule, an ID is automatically assigned in increments of 10 in the order in which rules are added. When more than one rule matches the command entered, the rule with the lowest ID takes precedence.

`platform <PLATFORM>`

Specifies the hardware or model details of the neighbor. Range: 1 to 128 alphanumeric characters.

`sw-version <SWVERSION>`

Specifies the software version of the neighbor. Range: 1 to 128 alphanumeric characters.

`voice-vlan-query <VLAN-ID>`

Specifies the VLAN query value of the neighbor. Range: 1 to 65535.

Authority

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

Examples

Adding rules to match a Cisco device with a specific software version on VLAN **512** to the CDP group **grp01**:

```
switch(config)# port-access cdp-group grp01
switch(config-cdp-group)# match platform CISCO
switch(config-cdp-group)# match sw-version 11.2(12)P
switch(config-cdp-group)# match voice-vlan-query 512
switch(config-cdp-group)# match seq 50 platform cisco sw-version 11.2(12)P voice-
vlan-query 512
switch(config-cdp-group)# exit
switch(config)# do show running-config

Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.000
!export-password: default
led locator on
!
!
vlan 1
port-access cdp-group grp01
     seq 10 match platform CISCO
     seq 20 match sw-version 11.2(12)P
     seq 30 match voice-vlan-query 512
     seq 50 match platform cisco sw-version 11.2(12)P voice-vlan-query 512
```

Removing a rule that matches the sequence number **25** from the CDP group named **grp01**:

```
switch(config)# port-access cdp-group grp01
switch(config-cdp-group)# no match seq 25
```

Adding a rule that matches the value of vendor-OUI **000b86** to the CDP group named **grp01**:

```
switch(config)# port-access cdp-group grp01
switch(config-cdp-group)# match vendor-oui 000b86
```

match (for LLDP groups)

Syntax

```
match [seq <SEQ-ID>] {sys-desc <SYS-DESC> | sysname <SYS-NAME> |
     vendor-oui <VENDOR-OUI> [type <KEY> [value <VALUE>]]}

no match [seq <SEQ-ID>] {sys-desc <SYS-DESC> | sysname <SYS-NAME> |
     vendor-oui <VENDOR-OUI> [type <KEY> [value <VALUE>]]}
```

Description

Defines a rule to match devices for an LLDP group. Up to 64 match/ignore rules can be defined for a group.

The `no` form of this command removes a rule.

Command context

`config-lldp-group`

Parameters

`seq <SEQ-ID>`

Specifies the ID of the rule to create or modify. If no ID is specified when adding a rule, an ID is automatically assigned in increments of 10 in the order in which rules are added. When more than one rule matches the command entered, the rule with the lowest ID takes precedence.

`sys-desc <SYS-DESC>`

Specifies the LLDP system description type-length-value (TLV). Range: 1 to 256 alphanumeric characters.

`sysname <SYS-NAME>`

Specifies the LLDP system name TLV. Range: 1 to 64 alphanumeric characters.

`vendor-oui <VENDOR-OUI>`

Specifies the LLDP system vendor OUI TLV. Range: 1 to 6 alphanumeric characters.

`type <KEY>`

Specifies the vendor OUI subtype key.

`value  <VALUE>`

Specifies the vendor OUI subtype value. Range: 1 to 256 alphanumeric characters.

Authority

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

Examples

Adding rules that match the LLDP system description **ArubaSwitch** and system name **Aruba** to the LLDP group named **grp01**:

```
switch(config)# port-access lldp-group grp01
switch(config-lldp-group)# match sys-desc ArubaSwitch
switch(config-lldp-group)# match sysname Aruba
switch(config)# do show running-config

Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.000
!export-password: default
led locator on
!
!
vlan 1
port-access lldp-group grp01
     seq 10 match sys-desc ArubaSwitch
     seq 20 match sysname Aruba
```

Removing a rule that matches the sequence number **25** from an LLDP group named **grp01**:

```
switch(config)# port-access lldp-group grp01
switch(config-lldp-group)# no match seq 25
```

Adding a rule that matches the value of vendor-OUI **000b86** with type of **1** to the LLDP group named **grp01**:

```
switch(config)# port-access lldp-group grp01
switch(config-lldp-group)# match vendor-oui 000b86 type 1
```

Adding a rule that matches the value of vendor-OUI **000c34** to the LLDP group named **grp01**:

```
switch(config)# port-access lldp-group grp01
switch(config-lldp-group)# match vendor-oui 000c34
```

match (for MAC groups)

Syntax

`[seq <SEQ-ID>] match {mac <MAC-ADDR> | mac-mask <MAC-MASK> | mac-oui <MAC-OUI>}`

`no [seq <SEQ-ID>] match {mac <MAC-ADDR> | mac-mask <MAC-MASK> | mac-oui <MAC-OUI>}`

Description

Defines a rule to match devices for a MAC group based on the criteria of MAC address, MAC address mask, or MAC Organizational Unique Identifier (OUI). Up to 64 match rules can be defined for a group.

You must not configure the following special MAC addresses:

- Null MAC—For example, 00:00:00:00:00:00 or 00:00:00/32
- Multicast MAC
- Broadcast MAC—For example, ff:ff:ff:ff:ff:ff
- System MAC

Although the switch accepts these addresses, it will not process these addresses for the local MAC match feature.

The `no` form of this command removes a rule for adding devices to a MAC group.

The number of clients that can onboard based on the match criteria is configured in the `aaa authentication port-access client-limit` command. For information about this command, see the *Security Guide* for your switch.

Command context

`config-mac-group`

Parameters

`seq <SEQ-ID>`

Specifies the entry sequence ID of the rule to create or modify a MAC group. If no ID is specified when adding a rule, an ID is automatically assigned in increments of 10 in the order in which rules are added. When more than one rule matches the command entered, the rule with the lowest ID takes precedence. Range: `1` to `4294967295`.

`mac <MAC-ADDR>`

Specifies the MAC address of the device.

`mac-mask <MAC-MASK>`

Specifies the MAC address mask to add devices in that range. Supported MAC address masks: `/32` and `/40`.

`mac-oui <MAC-OUI>`

Specifies the MAC OUI to add devices in that range. Supports MAC OUI address of maximum length of 24 bits.

Authority

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

Examples

Adding a device to the MAC group **grp01** based on complete MAC address:

```
switch(config)# mac-group grp01
switch(config-mac-group)# match mac 1a:2b:3c:4d:5e:6f
switch(config-mac-group)# exit
```

Adding devices to the MAC group **grp02** based on MAC mask:

```
switch(config)# mac-group grp01
switch(config-mac-group)# match mac-mask 1a:2b:3c:4d:5e/40
switch(config-mac-group)# match mac-mask 18:e3:ab:73/32
switch(config-mac-group)# exit
```

Adding devices to the MAC group **grp03** based on MAC OUI:

```
switch(config)# mac-group grp03
switch(config-mac-group)# match mac-oui 81:cd:93
switch(config-mac-group)# exit
```

Adding devices to the MAC group **grp01** with MAC entry sequence number and based on MAC address:

```
switch(config)# mac-group grp01
switch(config-mac-group)# seq 10 match mac b2:c3:44:12:78:11
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01
    seq 10 match mac b2:c3:44:12:78:11

```
```

Removing devices from the MAC group **grp01** based on sequence number:
```

```
switch(config)# mac-group grp01
switch(config-mac-group)# no match seq 10
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01

```

Adding devices to the MAC group **grp01** with MAC entry sequence number and based on MAC address, MAC address mask, and MAC OUI:

```
switch(config)# mac-group grp01
switch(config-mac-group)# seq 10 match mac b2:c3:44:12:78:11
switch(config-mac-group)# seq 20 match mac-oui 1a:2b:3c
switch(config-mac-group)# seq 30 match mac-mask 71:14:89:f3/32
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01
     seq 10 match mac b2:c3:44:12:78:11
     seq 20 match mac-oui 1a:2b:3c
     seq 30 match mac-mask 71:14:89:f3/32

```

Removing devices from the MAC group **grp01** based on MAC OUI:

```
switch(config)# mac-group grp01
switch(config-mac-group)# no seq 20 match mac-oui 1a:2b:3c
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
```

```
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp01
     seq 10 match mac b2:c3:44:12:78:11
     seq 30 match mac-mask 71:14:89:f3/32


```
```

Adding devices to the MAC group **grp03** with MAC entry sequence number and based on MAC address mask:

```
switch(config)# mac-group grp03
switch(config-mac-group)# seq 10 match mac-mask 10:14:a3:b7:55/40
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp03
     seq 10 match mac-mask 10:14:a3:b7:55/40

```
```

Removing devices from the MAC group **grp03** based on MAC address mask:

```
switch(config)# mac-group grp03
switch(config-mac-group)# no seq 10 match mac-mask 10:14:a3:b7:55/40
switch(config-mac-group)# exit
switch(config)# do show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.0001
!export-password: default
led locator on
!
!
vlan 1
interface mgmt
    no shutdown
    ip dhcp
mac-group grp03

```
```

port-access cdp-group

Syntax
```

```
port-access cdp-group <CDP-GROUP-NAME>

no port-access cdp-group <CDP-GROUP-NAME>
```

Description

Creates a CDP (Cisco Discovery Protocol) group or modifies an existing CDP group. A CDP Group is used to classify connected devices based on the CDP packet details advertised by the device. A maximum of 32 CDP groups can be configured on the switch. Each group accepts 64 match/ignore commands.

The `no` form of this command removes a CDP group.

Command context

`config`

Parameters

<CDP-GROUP-NAME>

Specifies the name of the CDP group to create or modify. The maximum number of characters supported is 32. Required.

Authority

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

Examples

Creating a CDP group named grp01:

```
switch(config)# port-access cdp-group grp01
switch(config-cdp-group)# match platform CISCO
switch(config-cdp-group)# match sw-version 11.2(12)P
switch(config-cdp-group)# match voice-vlan-query 512
switch(config-cdp-group)# seq 50 match platform cisco sw-version 11.2(12)P voice-
vlan-query 512
switch(config-cdp-group)# exit
switch(config)# do show running-config

Current configuration:
!
!Version ArubaOS-CX Virtual.10.0X.000
!export-password: default
led locator on
!
!
vlan 1
port-access cdp-group grp01
     seq 10 match platform CISCO
     seq 20 match sw-version 11.2(12)P
     seq 30 match voice-vlan-query 512
     seq 50 match platform cisco sw-version 11.2(12)P voice-vlan-query 512
```

Removing a CDP group named grp01:

```
switch(config)# no port-access cdp-group grp01
```

port-access device-profile

Syntax

```
port-access device-profile <DEVICE-PROFILE-NAME>

no port-access device-profile <DEVICE-PROFILE-NAME>
```

Description

Creates a new device profile and switches to the `config-device-profile` context. A maximum of 32 device profiles can be created.

The `no` form of this command removes a device profile.

Command context

`config`

Parameters

<DEVICE-PROFILE-NAME>

Specifies the name of a device profile. Range: 1 to 32 alphanumeric characters.

Authority

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

Examples

Creating a device profile named **profile01**:

```
switch(config)# port-access device-profile profile01
switch(config-device-profile)#
```

Removing a device profile named **profile01**:

```
switch(config)# no port-access device-profile profile01
```

port-access device-profile mode block-until-profile-applied

You must configure this mode in device profile only on standalone ports where there is no security configured and when you not want the port to be offline until one client is onboarded.

Syntax

```
port-access device-profile mode block-until-profile-applied

no port-access device-profile mode block-until-profile-applied
```

Description

Configures the switch to block the port until a profile match occurs for a device. This configuration is required when no security feature is enabled on the port.

You must enable this mode or security on the port for local MAC match feature to operate. You must not enable both features on the same port at the same time.

You must not combine any other AAA configurations with the block-until-profile-applied mode.

The `no` form of this command removes a rule for adding devices to a MAC group.

Command context

`config-if`

`config-if-deviceprofile`

Authority

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

Example

On the 6400 Switch Series, interface identification differs.

Configuring block-until-profile applied mode on port 1/1/1:

```
switch(config)# interface 1/1/1
switch(config-if)# port-access device-profile
switch(config-if-deviceprofile)# mode block-until-profile-applied
switch(config-if-deviceprofile)# end
```

port-access lldp-group

Syntax

`port-access lldp-group <LLDP-GROUP-NAME>`

`no port-access lldp-group <LLDP-GROUP-NAME>`

Description

Creates an LLDP group or modifies an existing LLDP group. An LLDP group is used to classify connected devices based on the LLDP type-length-values (TLVs) advertised by the device. A maximum of 32 LLDP groups can be configured on the switch. Each group accepts 64 match/ignore commands.

The `no` form of this command removes an LLDP group.

Command context

`config`

Parameters

<LLDP-GROUP-NAME>

Specifies the name of the LLDP group to create or modify. The maximum number of characters supported is 32. Required.

Authority

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

Examples

Creating an LLDP group named grp01:

```
switch(config)# port-access lldp-group grp01
switch(config-lldp-group)#
```

Removing an LLDP group named grp01:

```
switch(config)# no port-access lldp-group grp01
```

show port-access device-profile

Syntax

```
show port-access device-profile [[interface {all | <INTERFACE-ID>}
     [client-status <MAC-ADDR>]] | name <DEVICE-PROFILE-NAME>]
```

Description

Shows the client status for a specific MAC address or profile name.

Command context

Manager (#)

Parameters

`interface {all | <INTERFACE-ID>}`

Select `all` for all interfaces or specify the name of an interface in the format: `member/slot/port`.

`client-status <MAC-ADDR>`

Specifies a MAC address (`xx:xx:xx:xx:xx:xx`), where `x` is a hexadecimal number from 0 to F.

`name <DEVICE-PROFILE-NAME>`

Specifies the name of the device profile.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Showing the applied state of the device profiles:

```
switch# show port-access device-profile

    Profile Name    : accesspoints
    LLDP Groups     : 2920-grp
    CDP Groups      :
    MAC Groups      : 2920-mac-grp1,2920-iot-grp2
    Role            : local_role_1
    State           : Enabled

    Profile Name    : access_switches
    LLDP Groups     : 2920-grp
    CDP Groups      :
    MAC Groups      :
    Role            : local_2920_role
    State           : Enabled

    Profile Name    : iot_devices
    LLDP Groups     :
    CDP Groups      :
    MAC Groups      : iot_camera-grp1,iot_sensors-grp1
    Role            : local_2920_role
    State           : Enabled

    Profile Name    : lobbyaps
    LLDP Groups     :
    CDP Groups      : lobby_ap_cdp_grp
    MAC Groups      :
```

```
  Role            : test_ap_role
  State           : Disabled
```

Showing the applied state of the device profile on interface 1/1/3:

```
switch# show port-access device-profile interface 1/1/3 client-status
00:0c:29:9e:d1:20

Port 1/1/3, Neighbor-Mac  00:0c:29:9e:d1:20
    Profile Name    : lobbyaps
    LLDP Group      :
    CDP Group       : aruba-ap_cdp
    MAC Group       :
    Role            : test_ap_role
    Status          : Failed
    Failure Reason  : Failed to apply MAC based VLAN
```

Showing the applied state of a specific device profile:

```
switch# show port-access device-profile name lldp-group

    Profile Name            : lldp-group
    LLDP Groups             :
    CDP Groups              :
    MAC Groups              : pc-behind-phone, lldp
    Role                    : auth_role
    State                   : Enabled
```

LLDP

The IEEE 802.1AB Link Layer Discovery Protocol (LLDP) provides a standards-based method for network devices to discover each other and exchange information about their capabilities. An LLDP device advertises itself to adjacent (neighbor) devices by transmitting LLDP data packets on all interfaces on which outbound LLDP is enabled, and reading LLDP advertisements from neighbor devices on ports on which inbound LLDP is enabled. Inbound packets from neighbor devices are stored in a special LLDP MIB (management information base). This information can then be queried by other devices through SNMP.

LLDP information is used by network management tools to create accurate physical network topologies by determining which devices are neighbors and through which interfaces they connect. LLDP operates at layer 2 and requires an LLDP agent to be active on each interface that sends and receives LLDP advertisements. LLDP advertisements can contain a variable number of TLV (type, length, value) information elements. Each TLV describes a single attribute of a device such as: system capabilities, management IP address, device ID, port ID.

Packet boundaries

When multiple LLDP devices are directly connected, an outbound LLDP packet travels only to the next LLDP device. An LLDP-capable device does not forward LLDP packets to any other devices, regardless of whether they are LLDP-enabled.

An intervening hub or repeater forwards the LLDP packets it receives in the same manner as any other multicast packets it receives. Therefore, two LLDP switches joined by a hub or repeater handle LLDP traffic in the same way that they would if directly connected.

Any intervening 802.1D device or Layer-3 device that is either LLDP-unaware or has disabled LLDP operation drops the packet.

LLDP-MED

LLDP-MED (ANSI/TIA-1057/D6) extends the LLDP (IEEE 802.1AB) industry standard to support advanced features on the network edge for Voice Over IP (VoIP) endpoint devices with specialized capabilities and LLDP-MED standards-based functionality. LLDP-MED in the switches uses the standard LLDP commands described earlier in this section, with some extensions, and also introduces new commands unique to LLDP-MED operation. The show commands described elsewhere in this section are applicable to both LLDP and LLDP-MED operation. LLDP-MED enables:

- Configure Voice VLAN and advertise it to connected MED endpoint devices.
- Power over Ethernet (PoE) status and troubleshooting support via SNMP.

LLDP agent

When you enable LLDP on the switch, it is automatically enabled on all data plane interfaces. You can customize this behavior by manually enabling/disabling support on each interface.

Supported standards

The LLDP agent supports the following standards: IEEE 802.1AB-2005, Station, and Media Access Control Connectivity Discovery.

Supported interfaces

LLDP is supported on interfaces mapped to a physical port, and the Out-Of-Band Management (OOBM) port. It is not supported on logical interfaces, such as loopback, tunnels, and SVIs.

Operating modes

When LLDP is enabled, the switch periodically transmits an LLDP advertisement (packet) out each active port enabled for outbound LLDP transmissions and receives LLDP advertisements on each active port enabled to receive LLDP traffic.

The LLDP agent can operate in one of the following modes:

- Transmit and receive (TxRx): This is the default setting on all ports. It enables a given port to both transmit and receive LLDP packets and to store the data from received (inbound) LLDP packets in the switch's MIB.
- Transmit only (Tx): Enables a port to transmit LLDP packets that can be read by LLDP neighbors. However, the port drops inbound LLDP packets from LLDP neighbors without reading them. This prevents the switch from learning about LLDP neighbors on that port.
- Receive only (Rx): Enables a port to receive and read LLDP packets from LLDP neighbors and to store the packet data in the switch's MIB. However, the port does not transmit outbound LLDP packets. This prevents LLDP neighbors from learning about the switch through that port.
- Disabled: Disables LLDP packet transmissions and reception on a port. In this state, the switch does not use the port for either learning about LLDP neighbors or informing LLDP neighbors of its presence.

An LLDP agent operating in TxRx mode or Tx mode sends LLDP frames to its directly connected devices both periodically and when the local configuration changes.

Sending LLDP frames

Each time the LLDP operating mode of an LLDP agent changes, its LLDP protocol state machine reinitializes. A configurable reinitialization delay prevents frequent initializations caused by frequent changes to the operating mode. If you configure the reinitialization delay, an LLDP agent must wait the specified amount of time to initialize LLDP after the LLDP operating mode changes.

Receiving LLDP frames

An LLDP agent operating in TxRx mode or Rx mode confirms the validity of TLVs carried in every received LLDP frame. If the TLVs are valid, the LLDP agent saves the information and starts an aging timer. The initial value of the aging timer is equal to the TTL value in the Time To Live TLV carried in the LLDP frame. When the LLDP agent receives a new LLDP frame, the aging timer restarts. When the aging timer decreases to zero, all saved information ages out.

TLV support

By default, the agent sends and receives the following mandatory TLVs on each interface:

- Port ID
- Chassis ID
- TTL

By default, the following ANSI/TIA-1057 TLVs for LLDP Media Endpoint Discovery (MED) are enabled on an agent. Sending them depends on the configuration and reception of any MED TLVs:

- MAC/PHY status. Includes the bit rate and auto negotiation status of the link.
- Power Via MDI: Includes Power Over Ethernet related information for supported interfaces.
- Port description
- System name
- System description
- Management address
- System capabilities
- Port VLAN ID

By default, the agent sends and receives the following ANSI/TIA-1057 TLVs for LLDP Media Endpoint Discovery (MED):

- Capabilities: Indicates MED TLV capability.
- Power Via MDI: Includes Power Over Ethernet related information.
- Network Policy: Includes the VLAN configuration for voice application.
- Location: Location identification information.
- Extended Power Via MDI: Power Over Ethernet related information

TLV advertisements

The LLDP agent transmits the following:

- Chassis-ID: Base MAC address of the switch.
- Port-ID: Port number of the physical port.
- Time-to-Live (TTL): Length of time an LLDP neighbor retains advertised data before discarding it.
- System capabilities: Identifies the primary switch capabilities (bridge, router). Identifies the primary switch functions that are enabled, such as routing.
- System description: Includes switch model name and running software version, and ROM version.

- System name: Name assigned to the switch.
- Management address: Default address selection method unless an optional address is configured.
- Port description: Physical port identifier.
- Port VLAN ID: On an L2 port, contains access or native VLAN ID. On an L3 port, contains a value of 0. Trunk allowed VLANs information are not advertised as part of the Port VLAN ID TLV. (Not supported on the OOBM interface)

LLDP MED support

LLDP-MED interoperates with directly connected IP telephony (endpoint) clients and provides the following features:

- Advertisement of the voice VLAN configured on the interface which is used by connected IP telephony (endpoint) clients.
- Advertisement of the configured location on the switch that can be used by the connected endpoint.
- Support for the fast-start capability

LLDP-MED is intended for use with VoIP endpoints and is not designed to support links between network infrastructure devices (such as switch-to-switch or switch-to-router links).

Configuring the LLDP agent

Procedure

1. By default, the LLDP agent is enabled on all active interfaces. If LLDP was disabled, enable it with the command `lldp`.
2. By default, the LLDP agent transmits and receive on all interfaces. To customize LLDP behavior on a specific interface, use the commands `lldp transmit` and `lldp receive`.
3. By default, the LLDP agent sets the management address in all TLVs in the following order:
   a. LLDP management IP address.
   b. Loopback interface IP.
   c. ROP (L3 ports) or SVI (L2 ports).
   d. OOBM (Management interface IP).
   e. Base MAC.

   On the OOBM port, the following order is used:
   a. LLDP management IP address,
   b. IP address of the management interface (OOBM port).
   c. IP address of the loopback interface.
   d. Base MAC address of the switch.

   To specify a different address, use the commands `lldp management-ipv4-address` and `lldp management-ipv6-address`
4. By default, all supported TLVs are sent and received. To customize the list, use the command `lldp select-tlv`.
5. By default, support for the LLDP-MED TLV is enabled. To customize settings, use the commands `lldp med` and `lldp med-location`.
6. If required, adjust LLDP timer, holdtime, reinitialization delay, and transmit delay from their default values with the commands `lldp timer`, `lldp holdtime`, `lldp reinit`, and `lldp txdelay`.

Example

On the 6400 Switch Series, interface identification differs.

This example creates the following configuration:

- Enables LLDP support.
- Disables LLDP transmission on interface **1/1/1**.

```
switch(config)# lldp
switch(config)# interface 1/1/1
switch(config-copp)# no lldp transmit
```

LLDP commands

clear lldp neighbors

Syntax

```
clear lldp neighbors
```

Description

Clears all LLDP neighbor details.

Command context

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

Clearing all LLDP neighbor details:

```
switch# clear lldp neighbors
```

clear lldp statistics

Syntax

```
clear lldp statistics
```

Description

Clears all LLDP neighbor statistics.

Command context

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

Clearing all LLDP neighbor statistics:

```
switch# clear lldp statistics
```

lldp

Syntax

```
lldp

no lldp
```

Description

Enables LLDP support globally on all active interfaces. By default, LLDP is enabled.

The `no` form of this command disables LLDP support globally on all active interfaces. It does not remove any LLDP configuration settings.

Command context

`config`

Authority

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

Examples

Enabling LLDP:

```
switch(config)# lldp
```

Disabling LLDP:

```
switch(config)# no lldp
```

lldp dot3

Syntax

```
lldp dot3 {poe | macphy}

no lldp dot3 {poe | macphy}
```

Description

Sets the 802.3 TLVs to be advertised. By default, advertisement of both POE and MAC/PHY TLVs is enabled. Not supported on the OOBM interface.

The `no` form of this command disables advertisement of 802.3 TLVs.

Command context

`config-if`

Parameters

`poe`

Specifies advertisement of power over Ethernet data link classification.

`macphy`

Specifies advertisement of media access control and physical layer information.

Authority

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

Examples

Enabling advertisement of the POE TLV:

```
switch(config-if)# lldp dot3 poe
```

Disabling advertisement of the POE TLV:

```
switch(config-if)# no lldp dot3 poe
```

lldp holdtime

Syntax

`lldp holdtime <TIME>`

`no lldp holdtime`

Description

Sets the holdtime that is used to calculate the LLDP Time-to-Live value. Time-to-Live defines the length of time that neighbors consider LLDP information sent by this agent as valid. When Time-to-Live expires, the information is deleted by the neighbor. Time-to-live is calculated by multiplying holdtime by the value of `lldp timer`.

The `no` form of this command sets the holdtime to its default value of 4.

Command context

`config`

Parameters

`<TIME>`

Specifies the holdtime in seconds. Range: 2 to 10. Default: 4.

Authority

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

Examples

Setting the holdtime to 8 seconds:

```
switch(config)# lldp holdtime 8
```

Setting the holdtime to the default value of 4 seconds:

```
switch(config)# no lldp holdtime
```

lldp management-ipv4-address

Syntax

```
lldp management-ipv4-address <IPV4-ADDR>

no lldp management-ipv4-address
```

Description

Defines the IPv4 management address of the switch which is sent in the management address TLV. One IPv4 and one IPv6 management address can be configured.

If you do not define an LLDP management address, then LLDP uses one of the following (in order):

- IP address of the port
- IP address of the management interface
- Base MAC address of the switch

The `no` form of this command removes the IPv4 management address of the switch.

Command context

```
config
```

Parameters

<IPV4-ADDR>

Specifies the management address of the switch as an IPv4 format (`x.x.x.x`), where `x` is a decimal value from 0 to 255.

Authority

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

Examples

Setting the management address to **10.10.10.2**:

```
switch(config)# lldp management-ipv4-address 10.10.10.2
```

Removing the management address:

```
switch(config)# no lldp management-ipv4-address
```

lldp management-ipv6-address

Syntax

```
lldp management-ipv6-address <IPV6-ADDR>

no lldp management-ipv6-address
```

Description

Defines the IPv6 management address of the switch. The management address is encapsulated in the management address TLV.

If you do not define an LLDP management address, then LLDP uses one of the following (in order):

- IP address of the port
- IP address of the management interface
- Base MAC address of the switch

The `no` form of this command removes the IPv6 management address of the switch.

Command context

`config`

Parameters

`<IPV6-ADDR>`

Specifies an IP address in IPv6 format (`xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx`), where `x` is a hexadecimal number from 0 to F.

Authority

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

Examples

Setting the management address to 2001:db8:85a3::8a2e:370:7334:

```
switch(config)# lldp management-ipv6-address 2001:0db8:85a3::8a2e:0370:7334
```

Removing the management address:

```
switch(config)# no lldp management-ipv6-address
```

lldp med

Syntax

```
lldp med [poe [priority-override] | capability | network-policy]

no med [poe [priority-override] | capability | network-policy]
```

Description

Configures support for the LLDP-MED TLV. LLDP-MED (media endpoint devices) is an extension to LLDP developed by TIA to support interoperability between VoIP endpoint devices and other networking end-devices. The switch only sends the LLDP MED TLV after receiving a MED TLV from and connected endpoint device.

Not supported on the OOBM interface.

The `no` form of this command disables support for the LLDP MED TLV.

Command context

`config-if`

Parameters

`poe [priority-override]`

Specifies advertisement of power over Ethernet data link classification. The `priority-override` option overrides user-configured port priority for Power over Ethernet. When both `lldp dot3 poe` and `lldp med poe` are enabled, the `lldp dot3 poe3` setting takes precedence. Default: enabled.

`capability`

Specifies advertisement of supported LLDP MED TLVs. The capability TLV is always sent with other MED TLVs, therefore it cannot be disabled when other MED TLVs are enabled. Default: enabled.

`network-policy`

Network policy discovery lets endpoints and network devices advertise their VLAN IDs, and IEEE 802.1p (PCP and DSCP) values for voice applications. This TLV is only sent when a voice VLAN policy is present. Default: enabled.

Authority

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

Examples

Enabling advertisement of the network policy TLV:

```
switch(config-if)# lldp med network-policy
```

Disabling advertisement of the network policy TLV:

```
switch(config-if)# no lldp med network-policy
```

lldp med-location

Syntax

```
lldp med-location {civic-addr  | elin-addr }

no med-location {civic-addr | elin-addr }
```

Description

Configures support for the LLDP-MED TLV. Supports only civic address and emergency location information number (ELIN). Coordinate-based location is not supported.

The `no` form of this command disables support for the LLDP MED TLV.

Command context

`config-if`

Parameters

`civic-addr`

Configures the LLDP MED civic location TLV.

`elin-addr`

Configures support for the LLDP MED emergency location TLV.

Authority

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

Examples

Enabling support for the LLDP MED emergency location TLV:

```
switch(config-if)# lldp med-location elin-addr gher
```

Disabling support for the LLDP MED emergency location TLV:

```
switch(config-if)# no lldp med-location elin-addr gher
```

Enabling support for the LLDP MED civic address TLV:

```
switch(config-if)# lldp med-location civic-addr US 1 4 ret 6 tyu 7 tiyuo
```

Disabling support for the LLDP MED civic address TLV:

```
switch(config-if)# no lldp med-location civic-addr US 1 4 ret 6 tyu 7 tiyuo
```

lldp receive

Syntax

```
lldp receive

no lldp receive
```

Description

Enables reception of LLDP information on an interface. By default, LLDP reception is enabled on all active interfaces, including the OOBM interface.

The `no` form of this command disables reception of LLDP information on an interface.

Command context

`config-if`

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Enabling LLDP reception on interface **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# lldp receive
```

Disabling LLDP reception on interface **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# no lldp receive
```

Enabling LLDP reception on the OOBM interface:

```
switch(config)# interface mgmt
switch(config-if)# lldp receive
```

Disabling LLDP reception on the OOBM interface:

```
switch(config)# interface mgmt
switch(config-if)# no lldp receive
```

lldp reinit

Syntax

```
lldp reinit <TIME>

no lldp reinit
```

Description

Sets the amount of time (in seconds) to wait before performing LLDP initialization on an interface.

The `no` form of this command sets the reinitialization time to its default value of 2 seconds.

Command context

`config`

Parameters

<TIME>

Specifies the reinitialization time in seconds. Range: 1 to 10. Default: 2 seconds.

Authority

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

Examples

Setting the reinitialization time to 5 seconds:

```
switch(config)# lldp reinit 5
```

Setting the reinitialization time to the default value of 2 seconds:

```
switch(config)# no lldp reinit
```

lldp select-tlv

Syntax

```
lldp select-tlv <TLV-NAME>

no lldp select-tlv <TLV-NAME>
```

Description

Selects a TLV that the LLDP agent will send and receive. By default, all supported TLVs are sent and received.

The `no` form of this command stops the LLDP agent from sending and receiving a specific TLV.

Command context

`config`

Parameters

`select-tlv <TLV-NAME>`

Specifies the TLV name to send. The following TLV names are supported:

- `management-address`: Selected as follows:
  1. IPv4 or IPV6 management address.
  2. IP address of the lowest configured loopback interface.
  3. If layer 3, then the route-only port IP address. If layer 2, the IP address of the SVI.
  4. OOBM interface IP address.
  5. Base MAC address of the switch.
- `port-description`: A description of the port.
- `port-vlan-id`: VLAN ID assigned to the port.
- `system-capabilities`: Identifies the primary switch functions that are enabled, such as routing.
- `system-description`: Description of the system, comprised of the following information: hardware serial number, hardware revision number, and firmware version.
- `system-name`: Host name assigned to the switch.

Authority

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

Examples

Stopping the LLDP agent from sending the **port-description** TLV:

```
switch(config)# no lldp select-tlv port-description
```

Enabling the LLDP agent to send the **port-description** TLV:

```
switch(config)# lldp select-tlv port-description
```

lldp timer

Syntax

```
lldp timer <TIME>

no lldp timer
```

Description

Sets the interval (in seconds) at which local LLDP information is updated and TLVs are sent to neighboring network devices by the LLDP agent. The minimum setting for this timer must be four times the value of `lldp txdelay`.

For example, this is a valid configuration:

- `lldp timer` = 16
- `lldp txdelay` = 4

And, this is an invalid configuration:

- `lldp timer` = 5
- `lldp txdelay` = 2

When copying a saved configuration to the running configuration, the value for `lldp timer` is applied before the value of `lldp txdelay`. This can result in a configuration error if the saved configuration has a value of `lldp timer` that is not four times the value of `lldp txdelay` in the running configuration.
For example, if the saved configuration has the settings:

- `lldp timer` = 16
- `lldp txdelay` = 4

And the running configuration has the settings:

- `lldp timer` = 30
- `lldp txdelay` = 7

Then you will see an error indicating that certain configuration settings could not be applied, and you will have to manually adjust the value of `lldp txdelay` in the running configuration.

The `no` form of this command sets the update interval to its default value of 30 seconds.

Command context

`config`

Parameters

<TIME>

Specifies the update interval (in seconds). Range: 5 to 32768. Default: 30.

Authority

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

Examples

Setting the update interval to 7 seconds:

```
switch(config)# lldp timer 7
```

Setting the update interval to the default value of 30 seconds:

```
switch(config)# no lldp timer
```

lldp transmit

Syntax

`lldp transsmit`

`no lldp transmit`

Description

Enables transmission of LLDP information on specific interface. By default, LLDP transmission is enabled on all active interfaces, including the OOBM interface.

The `no` form of this command disables transmission of LLDP information on an interface.

Command context

`config-if`

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Enabling LLDP transmission on interface **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# lldp transsmit
```

Disabling LLDP transmission on interface **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# no lldp transsmit
```

Enabling LLDP transmission on the OOBM interface:

```
switch(config)# interface mgmt
switch(config-if)# lldp transsmit
```

Disabling LLDP transmission on the OOBM interface:

```
switch(config)# interface mgmt
switch(config-if)# no lldp transsmit
```

lldp txdelay

Syntax

`lldp txdelay <TIME>`

`no lldp txdelay`

Description

Sets the amount of time (in seconds) to wait before sending LLDP information from any interface. The maximum value for `txdelay` is 25% of the value of `lldp tx timer`.

The `no` form of this command sets the delay time to its default value of 2 seconds.

Command context

`config`

Parameters

<TIME>

Specifies the delay time in seconds. Range: 0 to 10. Default: 2.

Authority

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

Examples

Setting the delay time to 8 seconds:

```
switch(config)# lldp txdelay 8
```

Setting the delay time to the default value of 2 seconds:

```
switch(config)# no lldp txdelay
```

lldp trap enable

Syntax

```
lldp trap enable

no lldp trap enable
```

Description

Enables sending SNMP traps for LLDP related events from a particular interface. LLDP trap generation is enabled by default on all the interfaces and has to be disabled for interfaces on which traps are not required to be generated.

The `no` form of this command disables the LLDP trap generation.

LLDP trap generation is disabled by default at the global level and must be enabled before any LLDP traps are sent.

Command context

`config` and `config-if`

Authority

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

Examples

Enabling LLDP trap generation on global level:

```
switch(config)# lldp trap enable
```

Enabling LLDP trap generation on interface level:

```
switch(config-if)# lldp trap enable
```

Disabling LLDP trap generation on global level:

```
switch(config)# no lldp trap enable
```

Disabling LLDP trap generation on interface level:

```
switch(config-if)# no lldp trap enable
```

show lldp configuration

Syntax

```
show lldp configuration [<INTERFACE-ID>][vsx-peer]
```

Description

Shows LLDP configuration settings for all interfaces or a specific interface.

Command context

Manager (#)

Parameters

<INTERFACE-ID>

Specifies an interface. Format: `member/slot/port`.

`[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

On the 6400 Switch Series, interface identification differs.

Showing configuration settings for all interfaces:

```
switch# show lldp configuration

LLDP Global Configuration
=========================
LLDP Enabled                 : No
LLDP Transmit Interval       : 30
LLDP Hold Time Multiplier    : 4
LLDP Transmit Delay Interval : 2
LLDP Reinit Timer Interval   : 2
LLDP Trap Enabled            : No

TLVs Advertised
===============
Management Address
Port Description
Port VLAN-ID
System Description
System Name

LLDP Port Configuration
=======================
PORT           TX-ENABLED          RX-ENABLED          INTF-TRAP-ENABLED
```

```
--------------------------------------------------------------------------------
1/1/1          Yes                 Yes                 Yes
1/1/2          Yes                 Yes                 Yes
1/1/3          Yes                 Yes                 Yes
1/1/4          Yes                 Yes                 Yes
1/1/5          Yes                 Yes                 Yes
1/1/6          Yes                 Yes                 Yes
...........
...........
mgmt           Yes                 Yes                 Yes
```

This example shows configuration settings for interface **1/1/1**.

```
switch# show lldp configuration 1/1/1

LLDP Global Configuration
=========================
LLDP Enabled                 : Yes
LLDP Transmit Interval       : 30
LLDP Hold Time Multiplier    : 4
LLDP Transmit Delay Interval : 2
LLDP Reinit Timer Interval   : 2
LLDP Trap Enabled            : No


LLDP Port Configuration
=======================
PORT           TX-ENABLED          RX-ENABLED          INTF-TRAP-ENABLED
--------------------------------------------------------------------------------
1/1/1          Yes                 Yes                 Yes
```

show lldp configuration mgmt

Syntax

```
show lldp configuration mgmt
```

Description

Shows LLDP configuration settings for the OOBM interface.

Command context

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 configuration settings for all interfaces:

```
switch# show lldp configuration mgmt

LLDP Global Configuration
=========================
LLDP Enabled                 : Yes
```

```
LLDP Transmit Interval       : 30
LLDP Hold Time Multiplier    : 4
LLDP Transmit Delay Interval : 2
LLDP Reinit Timer Interval   : 2
LLDP Trap Enabled            : Yes


LLDP Port Configuration
=======================
PORT           TX-ENABLED          RX-ENABLED          INTF-TRAP-ENABLED
-------------------------------------------------------------------------
mgmt           Yes                 Yes                 Yes
```

show lldp local-device

Syntax

```
show lldp local-device[vsx-peer]
```

Description

Shows global LLDP information advertised by the switch, as well as port-based data. If VLANs are configured on any active interfaces, the VLAN ID is only shown for trunk native or untagged VLAN IDs on access interfaces.

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

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 global LLDP information only (all ports including OOBM port are administratively down):

```
switch# show lldp local-device

Global Data
===========

Chassis-ID             : 1c:98:ec:e3:45:00
System Name            : switch
System Description     : Aruba JL375A 8400X XL.01.01.0001
Management Address     : 192.168.10.1
Capabilities Available : Bridge, Router
Capabilities Enabled   : Bridge, Router
TTL                    : 120
```

Showing all ports except **1/1/11** and OOBM as administratively down:

```
switch# show lldp local-device

Global Data
===========

Chassis-ID              : 1c:98:ec:e3:45:00
System Name             : switch
System Description      : Aruba
Management Address      : 192.168.10.1
Capabilities Available : Bridge, Router
Capabilities Enabled    : Bridge, Router
TTL                     : 120

Port Based Data
===============

Port-ID           : 1/1/11
Port-Desc         : "1/1/11"
Port Mgmt-Address : 164.254.21.220
Port VLAN ID      : 0

Port-ID           : mgmt
Port-Desc         : "mgmt"
Port Mgmt-Address : 164.254.21.220
```

In this example, all the ports except **1/1/11** are administratively down, and VLAN ID 100 is configured on this access interface.

```
switch# show lldp local-device

Global Data
===========

Chassis-ID              : 1c:98:ec:e3:45:00
System Name             : switch
System Description      : Aruba
Management Address      : 192.168.10.1
Capabilities Available : Bridge, Router
Capabilities Enabled    : Bridge, Router
TTL                     : 120

Port Based Data
===============

Port-ID           : 1/1/11
Port-Desc         : "1/1/11"
Port VLAN ID      : 100
Parent Interface  : interface 1/1/11
```

show lldp neighbor-info

Syntax

```
show lldp neighbor-info [<INTERFACE-NAME>][vsx-peer]
```

Description

Displays information about neighboring devices for all interfaces or for a specific interface. The information displayed varies depending on the type of neighbor connected and the type of TLVs sent by the neighbor.

Command context

Manager (#)

Parameters

<INTERFACE-NAME>

Specifies the interface for which to show information for neighboring devices. Use the format `member/slot/port` (for example, `1/3/1`).

`[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

On the 6400 Switch Series, interface identification differs.

Showing LLDP information for all interfaces:

```
switch# show lldp neighbor-info

LLDP Neighbor Information
=========================

Total Neighbor Entries          : 3
Total Neighbor Entries Deleted  : 0
Total Neighbor Entries Dropped  : 0
Total Neighbor Entries Aged-Out : 0

LOCAL-PORT  CHASSIS-ID         PORT-ID       PORT-DESC     TTL      SYS-NAME
-------------------------------------------------------------------------------
1/1/1       70:72:cf:a4:7d:50  1/1/1         1/1/1         32       switch
1/1/2       48:0f:cf:af:73:80  1/1/2         1/1/2         120      switch
1/1/46      48:0f:cf:af:73:80  1/1/46        1/1/46        120      switch
mgmt        48:0f:cf:af:73:80  mgmt          mgmt          120      switch
```

Showing information for interface **1/3/1** when it has only one switch connected as a neighbor:

```
switch# show lldp neighbor-info 1/3/1

Port                           : 1/1/1
Neighbor Entries               : 1
Neighbor Entries Deleted       : 0
Neighbor Entries Dropped       : 0
Neighbor Entries Aged-Out      : 0
Neighbor Chassis-Name          : HP-3800-24G-PoEP-2XG
Neighbor Chassis-Description   : HP J9587A 3800-24G-PoE+-2XG Switch, revision...
Neighbor Chassis-ID            : 10:60:4b:39:3e:80
Neighbor Management-Address    : 192.168.1.1
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge
Neighbor Port-ID               : 1/1/1
Neighbor Port-Desc             : 1/1/1
Neighbor Port VLAN ID          :
TTL                            : 120
```

Showing information for interface **1/3/10** when the neighbor sends a DOT3 power TLV:

```
switch# show lldp neighbor-info 1/3/10
Port                           : 1/3/10
Neighbor Entries               : 1
Neighbor Entries Deleted       : 0
Neighbor Entries Dropped       : 0
Neighbor Entries Aged-Out      : 0
Neighbor Chassis-Name          : 84:d4:7e:ce:5d:68
Neighbor Chassis-Description   : ArubaOS (MODEL: 325), Version Aruba IAP
Neighbor Chassis-ID            : 84:d4:7e:ce:5d:68
Neighbor Management-Address    : 169.254.41.250
Chassis Capabilities Available : Bridge, WLAN
Chassis Capabilities Enabled   : WLAN
Neighbor Port-ID               : 84:d4:7e:ce:5d:68
Neighbor Port-Desc             : eth0
TTL                            : 120
Neighbor Port VLAN ID          :
Neighbor PoE information       : DOT3
Neighbor Power Type            : TYPE2 PD
Neighbor Power Priority        : Unkown
Neighbor Power Source          : Primary
PD Requested Power Value       : 25.0 W
PSE Allocated Power Value: 25.0 W
Neighbor Power Supported       : Yes
Neighbor Power Enabled         : Yes
Neighbor Power Class           : 5
Neighbor Power Paircontrol     : No
PSE Power Pairs                : Signal
```

Showing information for interface **1/1/1** when it has multiple neighbors (displays a maximum of four):

```
switch# show lldp neighbor-info 1/1/1

Port                           : 1/1/1
Neighbor Entries               : 4
Neighbor Entries Deleted       : 0
Neighbor Entries Dropped       : 0
Neighbor Entries Aged-Out      : 0
Neighbor Chassis-Name          : switch
Neighbor Chassis-Description   : Aruba JL375A 8400X XL.01.01.0001
Neighbor Chassis-ID            : 1c:98:ec:fe:25:00
Neighbor Management-Address    : 10.1.1.2
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID               : 1/1/1
Neighbor Port-Desc             : 1/1/1
Neighbor Port VLAN ID          :
TTL                            : 120
Neighbor Chassis-Name          : switch
Neighbor Chassis-Description   : Aruba JL375A 8400X XL.01.01.0001
Neighbor Chassis-ID            : 1c:98:ec:fe:25:01
Neighbor Management-Address    : 10.1.1.3
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID               : 1/1/1
Neighbor Port-Desc             : 1/1/1
Neighbor Port VLAN ID          :
TTL                            : 120
Neighbor Chassis-Name          : switch
Neighbor Chassis-Description   : Aruba JL375A 8400X XL.01.01.0001
```

```
Neighbor Chassis-ID            : 1c:98:ec:fe:25:02
Neighbor Management-Address    : 10.1.1.4
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID               : 1/1/1
Neighbor Port-Desc             : 1/1/1
Neighbor Port VLAN ID          : 50
TTL                            : 120
Neighbor Chassis-Name          : switch
Neighbor Chassis-Description   : Aruba JL375A 8400X XL.01.01.0001
Neighbor Chassis-ID            : 1c:98:ec:fe:25:03
Neighbor Management-Address    : 10.1.1.5
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID               : 1/1/1
Neighbor Port-Desc             : 1/1/1
Neighbor Port VLAN ID          : 100
TTL                            : 120
```

Showing neighbor information for interface 1/3/2 when it has EEE enabled and successfully auto-negotiated:

```
switch# show lldp neighbor-info 1/3/2

Port                           : 1/3/2
Neighbor Entries               : 1
Neighbor Entries Deleted       : 1
Neighbor Entries Dropped       : 0
Neighbor Entries Aged-Out      : 1
Neighbor Chassis-Name          : BLDG01-F1-6300
Neighbor Chassis-Description   : Aruba JL668A  FL.10.07.0001BN
Neighbor Chassis-ID            : 88:3a:30:92:a5:c0
Neighbor Management-Address    : 10.6.9.15
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID               : 1/1/1
Neighbor Port-Desc             : 1/1/1
Neighbor Port VLAN ID          : 1
TTL                            : 120

Neighbor Mac-Phy details
Neighbor Auto-neg Supported    : true
Neighbor Auto-Neg Enabled      : true
Neighbor Auto-Neg Advertised   : 1000 BASE_TFD, 100 BASE_T4, 10 BASET_FD
Neighbor MAU type              : 1000 BASETFD

Neighbor EEE information       : DOT3
Neighbor TX Wake time          : 17 us
Neighbor RX Wake time          : 17 us
Neighbor Fallback time         : 17 us
Neighbor TX Echo time          : 17 us
Neighbor RX Echo time          : 17 us
```

show lldp neighbor-info detail

Syntax

```
show lldp neighbor-info detail [vsx-peer]
```

Description

Shows detailed LLDP neighbor information for all LLDP neighbor connected interfaces.

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

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

On the 6400 Switch Series, interface identification differs.

Showing detailed LLDP information for all interfaces:

```
switch# show lldp neighbor-info detail

LLDP Neighbor Information
=========================

Total Neighbor Entries          : 6
Total Neighbor Entries Deleted  : 2
Total Neighbor Entries Dropped  : 0
Total Neighbor Entries Aged-Out : 2

--------------------------------------------------------------------------------

Port                            : 1/1/1
Neighbor Entries                : 1
Neighbor Entries Deleted        : 0
Neighbor Entries Dropped        : 0
Neighbor Entries Aged-Out       : 0
Neighbor Chassis-Name           : 6300
Neighbor Chassis-Description    : Aruba ...
Neighbor Chassis-ID             : 38:11:17:1a:d5:00
Neighbor Management-Address     : 38:11:17:1a:d5:00
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled    : Bridge, Router
Neighbor Port-ID                : 1/1/4
Neighbor Port-Desc              : 1/1/4
Neighbor Port VLAN ID           : 1
TTL                             : 120

Neighbor Mac-Phy details
Neighbor Auto-neg Supported     : true
Neighbor Auto-Neg Enabled       : true
Neighbor Auto-Neg Advertised    : 1000 BASE_TFD, 100 BASE_T4, 10 BASET_FD
Neighbor MAU type               : 1000 BASETFD

--------------------------------------------------------------------------------

Port                            : 1/1/2
Neighbor Entries                : 1
Neighbor Entries Deleted        : 0
Neighbor Entries Dropped        : 0
```

```
Neighbor Entries Aged-Out      : 0
Neighbor Chassis-Name          : 6300
Neighbor Chassis-Description   : Aruba ...
Neighbor Chassis-ID            : 38:11:17:1a:d5:00
Neighbor Management-Address    : 38:11:17:1a:d5:00
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID               : 1/1/5
Neighbor Port-Desc             : 1/1/5
Neighbor Port VLAN ID          : 1
TTL                            : 120

Neighbor Mac-Phy details
Neighbor Auto-neg Supported    : true
Neighbor Auto-Neg Enabled      : true
Neighbor Auto-Neg Advertised   : 1000 BASE_TFD, 100 BASE_T4, 10 BASET_FD
Neighbor MAU type              : 1000 BASETFD

--------------------------------------------------------------------------------

Port                           : 1/1/3
Neighbor Entries               : 1
Neighbor Entries Deleted       : 0
Neighbor Entries Dropped       : 0
Neighbor Entries Aged-Out      : 0
Neighbor Chassis-Name          : 6300
Neighbor Chassis-Description   : Aruba ...
Neighbor Chassis-ID            : 38:11:17:1a:d5:00
Neighbor Management-Address    : 38:11:17:1a:d5:00
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID               : 1/1/6
Neighbor Port-Desc             : 1/1/6
Neighbor Port VLAN ID          : 1
TTL                            : 120

Neighbor Mac-Phy details
Neighbor Auto-neg Supported    : true
Neighbor Auto-Neg Enabled      : true
Neighbor Auto-Neg Advertised   : 1000 BASE_TFD, 100 BASE_T4, 10 BASET_FD
Neighbor MAU type              : 1000 BASETFD

--------------------------------------------------------------------------------

Port                           : 1/1/46
Neighbor Entries               : 1
Neighbor Entries Deleted       : 0
Neighbor Entries Dropped       : 0
Neighbor Entries Aged-Out      : 0
Neighbor Chassis-Name          : 6300
Neighbor Chassis-Description   : Aruba ...
Neighbor Chassis-ID            : 38:11:17:1a:d5:00
Neighbor Management-Address    : 38:11:17:1a:d5:00
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID               : 1/1/19
Neighbor Port-Desc             : 1/1/19
Neighbor Port VLAN ID          : 1
TTL                            : 120

Neighbor Mac-Phy details
Neighbor Auto-neg Supported    : true
```

```
Neighbor Auto-Neg Enabled      : true
Neighbor Auto-Neg Advertised   : 1000 BASE_TFD, 100 BASE_T4, 10 BASET_FD
Neighbor MAU type              : 1000 BASETFD

--------------------------------------------------------------------------------

Port                           : 1/1/47
Neighbor Entries               : 1
Neighbor Entries Deleted       : 0
Neighbor Entries Dropped       : 0
Neighbor Entries Aged-Out      : 0
Neighbor Chassis-Name          : 6300
Neighbor Chassis-Description   : Aruba ...
Neighbor Chassis-ID            : 38:11:17:1a:d5:00
Neighbor Management-Address    : 38:11:17:1a:d5:00
Chassis Cap
```

show lldp neighbor-info mgmt

Syntax

```
show lldp neighbor-info mgmt
```

Description

Displays information about neighboring devices connected to the OOBM interface.

Command context

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 LLDP information for the OOBM interface:

```
switch# show lldp neighbor-info mgmt

Port                           : mgmt
Neighbor Entries               : 1
Neighbor Entries Deleted       : 0
Neighbor Entries Dropped       : 0
Neighbor Entries Aged-Out      : 0
Neighbor Chassis-Name          : HP-3800-24G-PoEP-2XG
Neighbor Chassis-Description   : HP J9587A 3800-24G-PoE+-2XG Switch, revision...
Neighbor Chassis-ID            : 10:60:4b:39:3e:80
Neighbor Management-Address    : 192.168.1.1
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge
Neighbor Port-ID               : mgmt
Neighbor Port-Desc             : mgmt
Neighbor Port VLAN ID          :
TTL                            : 120
```

Showing LLDP information for the OOBM interface when there are four neighbors:

```
switch# show lldp neighbor-info mgmt

Port                            : mgmt
Neighbor Entries                : 4
Neighbor Entries Deleted        : 0
Neighbor Entries Dropped        : 0
Neighbor Entries Aged-Out       : 0
Neighbor Chassis-Name           : switch
Neighbor Chassis-Description    : Aruba JL375A 8400X XL.01.01.0001
Neighbor Chassis-ID             : 1c:98:ec:fe:25:00
Neighbor Management-Address     : 10.1.1.2
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID                : 1/1/1
Neighbor Port-Desc              : 1/1/1
Neighbor Port VLAN ID           :
TTL                             : 120

Neighbor Chassis-Name           : switch
Neighbor Chassis-Description    : Aruba JL375A 8400X XL.01.01.0001
Neighbor Chassis-ID             : 1c:98:ec:fe:25:01
Neighbor Management-Address     : 10.1.1.3
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID                : 1/1/1
Neighbor Port-Desc              : 1/1/1
Neighbor Port VLAN ID           :
TTL                             : 120

Neighbor Chassis-Name           : switch
Neighbor Chassis-Description    : Aruba JL375A 8400X XL.01.01.0001
Neighbor Chassis-ID             : 1c:98:ec:fe:25:02
Neighbor Management-Address     : 10.1.1.4
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID                : 1/1/1
Neighbor Port-Desc              : 1/1/1
Neighbor Port VLAN ID           :
TTL                             : 120

Neighbor Chassis-Name           : switch
Neighbor Chassis-Description    : Aruba JL375A 8400X XL.01.01.0001
Neighbor Chassis-ID             : 1c:98:ec:fe:25:03
Neighbor Management-Address     : 10.1.1.5
Chassis Capabilities Available : Bridge, Router
Chassis Capabilities Enabled   : Bridge, Router
Neighbor Port-ID                : 1/1/1
Neighbor Port-Desc              : 1/1/1
Neighbor Port VLAN ID           :
TTL                             : 120
```

show lldp statistics

Syntax

`show lldp statistics [<INTERFACE-ID>][vsx-peer]`

Description

Shows global LLDP statistics or statistics for a specific interface.

Command context

Manager (#)

Parameters

<INTERFACE-ID>

Specifies an interface. Format: `member/slot/port`.

`[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

On the 6400 Switch Series, interface identification differs.

Showing global statistics for all interfaces:

```
switch# show lldp statistics
LLDP Global Statistics
======================

Total Packets Transmitted            : 19
Total Packets Received               : 19
Total Packets Received And Discarded : 0
Total TLVs Unrecognized              : 0

LLDP Port Statistics
====================

PORT-ID        TX-PACKETS     RX-PACKETS     RX-DISCARDED   TLVS-UNKNOWN
-------------------------------------------------------------------------
1/1/1          7              7              0              0
1/1/2          7              7              0              0
1/1/3          0              0              0              0
1/1/4          0              0              0              0
1/1/5          0              0              0              0
...
mgmt           5              5              0              0

```
```

Showing statistics for interface **1/1/1**:

```
switch# show lldp statistics 1/1/1

LLDP Statistics
===============

Port Name                          : 1/1/1
Packets Transmitted                : 159
Packets Received                   : 163
Packets Received And Discarded     : 0
Packets Received And Unrecognized  : 0
```

show lldp statistics mgmt
```

Syntax

```
show lldp statistics mgmt
```

Description

Shows LLDP statistics for the OOBM interface.

Command context

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 LLDP statistics for the OOBM interface:

```
switch# show lldp statistics mgmt

LLDP Statistics
===============

Port Name                          : mgmt
Packets Transmitted                : 20
Packets Received                   : 23
Packets Received And Discarded     : 0
Packets Received And Unrecognized : 0
```

show lldp tlv

Syntax

```
show lldp tlv[vsx-peer]
```

Description

Shows the LLDP TLVs that are configured for send and receive.

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

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# show lldp tlv

TLVs Advertised
===============

Management Address
Port Description
Port VLAN-ID
System Capabilities
System Description
System Name
```

Cisco Discovery Protocol (CDP)

Cisco Discovery Protocol (CDP) is a proprietary layer 2 protocol supported by most Cisco devices. It is used to exchange information, such as software version, device capabilities, and voice VLAN information, between directly connected devices, such as a VoIP phone and a switch.

CDP support

By default, CDP is enabled on each active switch port. This is a read-only capability, which means the switch can receive and store information about adjacent CDP devices, but does not generate CDP packets (except when communicating with Cisco IP phones.)

The switch supports CDPv2 only and does not support SNMP MIB traps.

When a CDP-enabled port receives a CDP packet from another CDP device, it enters data for that device into the CDP Neighbors table, along with the port number on which the data was received. It does not forward the packet. The switch also periodically purges the table of any entries that have expired. (The holdtime for any data entry in the switch CDP Neighbors table is configured in the device transmitting the CDP packet and cannot be controlled in the switch receiving the packet.) A switch reviews the list of CDP neighbor entries every three seconds and purges any expired entries.

Support for legacy Cisco IP phones

Autoconfiguration of legacy Cisco IP phones for tagged voice VLAN support requires CDPv2.

On initial boot-up, and sometimes periodically, a Cisco phone queries the switch and advertises information about itself using CDPv2. When the switch receives the VoIP VLAN Query TLV (type 0x0f) from the phone, the switch immediately responds with the voice VLAN ID in a reply packet using the VoIP VLAN Reply TLV (type 0x0e). This enables the Cisco phone to boot properly and send traffic on the advertised voice VLAN ID.

The switch CDP packet includes these TLVs:

- CDP Version: 2
- CDP TTL: 180 seconds
- Checksum
- Capabilities (type 0x04): 0x0008 (is a switch)
- Native VLAN: The PVID of the port
- VoIP VLAN Reply (type 0xe): voice VLAN ID (same as advertised by LLDP-MED)
- Trust Bitmap (type 0x12): 0x00
- Untrusted port CoS (type 0x13): 0x00

CDP commands

cdp

Syntax

```
cdp
```

Description

Configures CDP support globally on all active interfaces or on a specific interface. By default, CDP is enabled on all active interfaces.

When CDP is enabled, the switch adds entries to its CDP Neighbors table for any CDP packets it receives from neighboring CDP devices.

When CDP is disabled, the CDP Neighbors table is cleared and the switch drops all inbound CDP packets without entering the data in the CDP Neighbors table.

The `no` form of this command disables CDP support globally on all active interfaces or on a specific interface.

Command context

```
config
config-if
```

Authority

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

Examples

Enabling CDP globally:

```
switch(config)# cdp
```

Disabling CDP globally:

```
switch(config)# no cdp
```

Enabling CDP on interface **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# cdp
```

Disabling CDP on interface **1/1/1**:

```
switch(config)# interface 1/1/1
switch(config-if)# no cdp
```

clear cdp counters

Syntax

```
clear cdp counters
```

Description

Clears CDP counters.

Command context

```
config
```

Authority

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

Examples

Clearing CDP counters:

```
switch(config) clear cdp counters
```

clear cdp neighbor-info

Syntax

```
clear cdp neighbor-info
```

Description

Clears CDP neighbor information.

Command context

```
config
```

Authority

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

Examples

Clearing CDP neighbor information:

```
switch(config) clear neighbor-info
```

show cdp

Syntax

```
show cdp
```

Description

Shows CDP information for all interfaces.

Command context

```
config
```

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Showing CDP information:

```
switch(config)# show cdp
CDP Global Information
```

```
======================
CDP           : Enabled
CDP Mode      : Rx only
CDP Hold Time : 180 seconds

Port         CDP
--------  --------------
1/1/1          Enabled
1/1/2          Enabled
1/1/3          Enabled
1/1/4          Enabled
1/1/5          Enabled
1/1/6          Enabled
1/1/7          Enabled
1/1/8          Enabled
1/1/9          Enabled
1/1/10         Enabled
```

show cdp neighbor-info

Syntax

`show cdp neighbor-info <INTERFACE-ID>`

Description

Shows CDP information for all neighbors or for CDP information on a specific interface.

Command context

`config`

Parameters

<INTERFACE-ID>

Specifies an interface. Format: `member/slot/port`.

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Showing all CDP neighbor information:

```
switch(config)# show cdp neighbor-info
Port        Device ID               Platform                Capability
-------------------------------------------------------------------------------
1/1/1       myswitch                cisco WS-C2950-12       SI
```

Showing CDP information for interface **1/1/1**:

```
switch(config)# show cdp neighbor-info 1/1/1
Local Port : 1/1/1
MAC                : 3c:a8:2a:7b:6b:2b
Device ID          : SEPd4adbd2a30d6
Address            : 2.71.0.230
```

```
Platform           : Cisco IP Phone 3905
Duplex             : full
Capability         : host
Voice VLAN Support : Yes
Neighbor Port-ID   : Port 1
```

show cdp traffic

Syntax

```
show cdp neighbor-info
```

Description

Shows CDP statistics for each interface.

Command context

```
config
```

Authority

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

Examples

On the 6400 Switch Series, interface identification differs.

Showing CDP traffic statistics:

```
switch(config)# show cdp traffic
CDP Statistics
====================
Port         Transmitted Frames       Received Frames          Discarded Frames
-------------------------------------------------------------------------------
1/1/1           0                          4                          0
1/1/2           0                          0                          0
1/1/3           0                          2                          0
1/1/4           0                          0                          0
1/1/5           0                          0                          0
```

Zero Touch Provisioning (ZTP) enables the auto-configuration of factory default switches without a network administrator onsite.

When a switch is booted from its factory default configuration, ZTP autoprovisions the switch by automatically downloading and installing a firmware file, a configuration file, or both. With ZTP, even a nontechnical user (for example: a store manager in a retail chain or a teacher in a school) can deploy devices at a site.

ZTP support

The switch supports standards-based Zero Touch Provisioning (ZTP) operations as follows:

- The switch must be running the factory default configuration.
- The switch can connect to the DHCP server from the OOBM management port.

  The switch can connect to the DHCP server from either the OOBM management port, or a data port on the default VLAN.
- ZTP operations are supported over IPv4 connections only. IPv6 connections are not supported for ZTP operations.
- You must configure the DHCP server to provide a standards-based ZTP server solution. Options and features that are specific to Network Management Solution (NMS) tools, such as AirWave, are not supported.
  - Aruba Central on-premise can manage AOS-CX switches on supported models through DHCP ZTP using two approaches:
    - On the DHCP server, configure DHCP option-60 as "ArubaInstantAP" 90 and provide the value in option-43 in the format *<group-details>, <aruba-central-on-prem-ip-or-fqdn>, <shared-secret>.*
    - On the DHCP server, configure DHCP option-60 as HPE vendor VCI and provide the value in option-43 in the tag-length-value (TLV) format with sub-option code of 146 as the Aruba Central on-premise FQDN or IPv4 address.
  - Supported DHCP options are:

DHCP option	Description
43	Vendor Specific Information
43 suboption 144	Name of the configuration file
43 suboption 145	Name of the firmware image file
43 suboption 146	Aruba Central FQDN or IPv4 address
43 suboption 148	HTTP Proxy FQDN or IPv4 address

DHCP option	Description
60	Vendor Class Identifier (VCI)
66	IPv4 address of the TFTP server (Specifying a host name instead of an IP address is not supported.)
67	Name of the configuration file (Option 43 suboption 144 takes precedence over this option.)

- The configuration file is a text file or JSON file that becomes the startup and running configuration on the switch after the ZTP operation is complete. The configuration can be in CLI or in JSON format.
- When the switch is started using the factory default configuration, the ZTP operation is started automatically and is active until any running configuration of the switch is modified. There is no CLI command required to start the operation.

The switch supports the following standards:

- RFC 2131, *Dynamic Host Configuration Protocol*.
- RFC 2132, *DHCP Options and BOOTP Vendor Extensions*. Support is limited to the options listed in the table "Supported DHCP options for ZTP on AOS-CX."

Hewlett Packard Enterprise recommends that you implement ZTP in a secure and private environment. Any public access can compromise the security of the switch, as follows:

- ZTP is enabled only in the factory default configuration of the switch, DHCP snooping is not enabled. The Rogue DHCP server must be manually managed.
- The DHCP offer is in plain data without encryption.

Setting up ZTP on a trusted network

The following procedure is an overview of setting up a Zero Touch Provisioning (ZTP) environment to provision newly installed switches automatically. The procedure is intended for network administrators who are familiar with automatically provisioning switches in a network, and does not provide detailed information about configuring or managing switches.

Procedure

1. For each switch model to be provisioned using ZTP, do the following:
   a. Obtain the switch firmware image file.
   b. Prepare the switch configuration file. The configuration file becomes the running configuration and the startup configuration on the switch.
2. Set up a TFTP server and record its IP address. The address is required when you set up the DHCP server. The switch must be able to reach the TFTP server and DHCP server, either on the same subnet, or on a remote subnet via DHCP relay.

   Switches support provisioning through a network connected to a data port or through a network connected to the management port.
3. Publish the configuration files and image files to the TFTP server. You need to know the locations of the files and the IP address of the TFTP server when you set up the vendor class options on the DHCP server.

4. On the DHCP server, set up vendor classes for each switch model you plan to provision. To do this you need the following information:
   - The IP address of the TFTP server. Using a host name is not supported.
   - The path to the switch configuration and firmware image files on the TFTP server.
   - The vendor class identifier (VCI) for each switch model.

   You can obtain the VCI by entering the `show dhcp client vendor-class-identifier` command from a switch CLI command prompt in the manager context. The VCI is the text string in the response that starts with `Aruba`.

   For example:

   ```
   switch# show dhcp client vendor-class-identifier
   Vendor Class Identifier:  Aruba xxxxx xxxx
   ```

   Where x indicates the switch model number.
5. At the installation site, provide the switch installer with a Cat6 network cable connected to the network that includes the DHCP and TFTP servers, and information about the switch port to use. The switch installer plugs the cable into the data port you specify.

   The ZTP operation begins when power is applied to the switch after the network cable is installed.
6. Assuming the downloaded configuration includes a way to access the CLI of the switch, you can enter the following command to show the options offered by the DHCP server and the status of the ZTP operation:

   `show ztp information`

ZTP process during switch boot

1. The switch boots up with the factory default configuration.

   If the ZTP operation detects that the switch configuration is different from the factory default configuration, the ZTP operation ends. The switch must be configured at the installation site.
2. The switch sends out a DHCP discovery from the management port.

   On switches that support ZTP operations on data ports, the switch also sends out a DHCP discovery from all data ports in the default VLAN.

   The switch waits to receive DHCP options indefinitely or until the running configuration is modified. If a DHCP IP address is received but no DHCP options are received, the switch waits an additional minute before ending the ZTP operation.

   On switches that support ZTP operations on data ports, DHCP options received on the management port have priority over DHCP options received on data ports:
   - If DHCP options are received on the management port before being received on a data port, the switch processes those options immediately.
   - If DHCP options are received on a data port, the switch waits an additional 30 seconds for options to be received on the management port. If no DHCP options are received on the management port during those 30 seconds, the switch processes the DHCP options it received on the data port.

   After the ZTP operation ends, there is no automatic retry. You can either attempt to boot the switch with the factory default configuration again, configure the switch at the installation site, or use the ZTP force-provision CLI to trigger the ZTP process, ignoring the present running configuration of the switch.

- Once force-provision is enabled, new DHCP requests are sent from the switch. Disabling force-provision does not stop the DHCP already in progress, but only changes the switch configuration status of force-provision.
- If ZTP fails while force-provision is enabled, there is no automatic retry. To retry, `ztp force-provision` should be disabled and re-enabled to clear the current ZTP state and send a new DHCP request. When `ztp force-provision` is already enabled on the switch, re-enabling it results in no operation.
- If the DHCP server is configured to provide both ZTP image and configuration options and there is a non-default startup configuration present on the switch, clearing the non-default startup configuration before triggering `ztp force-provision` is recommended. If an image is downloaded via ZTP, the switch reboots once the image download is complete and ZTP force-provision configuration is lost, causing ZTP to enter into a failed state. ZTP force-provision will need to be enabled again to continue the process.

3. The DHCP server responds with an offer containing the following:
   - The IPv4 address of the TFTP server
   - One or both of the following:
     - The name of the firmware image file
     - The name of the configuration file
   - Aruba Central Location (optional)
   - HTTP Proxy Location (optional)
4. If a firmware image file is offered, the ZTP operation downloads the image file from the TFTP server to the switch. If the current switch image and downloaded firmware image version do not match, then the switch boots with the downloaded image:
   - If the image upgrade fails, the switch retains its original firmware image and the ZTP operation ends with a failed status.
   - If the image upgrade succeeds, the ZTP operation is started again after the switch reboots. Because the downloaded image file matches the image file installed on the switch, the ZTP operation continues, and checks if a configuration file is offered.
5. If a configuration file is offered, the ZTP operation downloads the configuration file copies the file to the running-config and then to the startup-config of the switch:
   - If the startup configuration update fails, the switch retains its factory-default running configuration and the ZTP operation ends with a failed status.
     - If the copy operation fails, the ZTP operation ends with a failed status.
     - If the copy operation succeeds, the ZTP operation ends successfully.

ZTP VSF switchover support

ZTP status is not synced in the VSF stack. When the VSF stack is formed, configuration changes are applied on the master switch, which is then synced to standby switch. When the switchover is performed on the VSF stack, the standby becomes the new master switch.

As part of the switchover process, the ZTP daemon starts on the new master. The status of the ZTP is failed because there are configuration changes present.

ZTP commands

show ztp information

Syntax

```
show ztp information
```

Description

Shows information about Zero Touch Provisioning (ZTP) operations performed on the switch.

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.

Usage

When a switch configured to use ZTP is booted from a factory default configuration, the switch contacts a DHCP server, which offers options for obtaining files used to provision the switch:

- The IP address of the TFTP server
- The name of the image file
- The name of the configuration file

The `show ztp information` command shows the options offered by the DHCP server and the status of the ZTP operation.

The status of the ZTP operation is one of the following:

Success

The ZTP operation succeeded.

One of the following is true:

- Both the running configuration and the startup configuration were updated.
- The IP address of the TFTP server was received, but the offer did not include a configuration file or a firmware image file.
- Any combination of vendor encapsulated DHCP options are received as configured, along with the firmware image and switch configuration file.
- Only vendor encapsulated DHCP options are configured and are received accordingly.

Failed - Custom startup configuration detected

The switch was booted from a configuration that is not the factory default configuration. For example, the administrator password has been set.

Failed - Timed out while waiting to receive ZTP options

Either the switch received the DHCP IPv4 address but no ZTP options were received within 1 minute or ZTP force-provision is triggered and no ZTP options are received within 3 minutes.

Failed - Detected change in running configuration

The running configuration was modified by a user while the ZTP operation was in progress.

Failed - TFTP server unreachable

The TFTP server is not reachable at the specified IP address.

Failed - TFTP server information unavailable

The image file name or config file name is provided without the TFTP server location to fetch the files from and ZTP enters failed state.

Failed - Invalid configuration file received

Either the file transfer of the configuration file failed, or the configuration file is invalid (an error occurred while attempting to apply the configuration).

Failed - Invalid image file received

Either the file transfer of the firmware image file failed, or the firmware image file is invalid (an error occurred while verifying the image).

Examples

Showing switch image download in progress after receiving ZTP options:

```
switch# show ztp information
TFTP Server              : 10.0.0.2
Image File               : TL_10_02_0001.swi
Configuration File       : config_file
ZTP Status               : In-progress - Image download and verification
Aruba Central Location   : secure.arubanetworks.com
Force-Provision          : Disabled
HTTP Proxy Location      : http.proxy.arubanetworks.com
```

Showing switch image download failure after receiving ZTP options:

```
switch# show ztp information
TFTP Server              : 10.0.0.2
Image File               : TL_10_02_0001.swi
Configuration File       : config_file
ZTP Status               : Failed - Unable to download image
Aruba Central Location   : secure.arubanetworks.com
Force-Provision          : Disabled
HTTP Proxy Location      : http.proxy.arubanetworks.com
```

Showing switch configuration download in progress after receiving ZTP options:

```
switch# show ztp information
TFTP Server              : 10.0.0.2
Image File               : TL_10_02_0001.swi
Configuration File       : config_file
ZTP Status               : In-progress - Configuration download
Aruba Central Location   : secure.arubanetworks.com
Force-Provision          : Disabled
HTTP Proxy Location      : http.proxy.arubanetworks.com
```

Showing switch configuration download failure after receiving ZTP options:

```
switch# show ztp information
TFTP Server              : 10.0.0.2
Image File               : TL_10_02_0001.swi
Configuration File       : config_file
ZTP Status               : Failed - Unable to download configuration
Aruba Central Location   : secure.arubanetworks.com
Force-Provision          : Disabled
HTTP Proxy Location      : http.proxy.arubanetworks.com
```

Showing switch failure to update start-up confriguration after downloading configuration received from ZTP options:

```
switch# show ztp information
TFTP Server              : 10.0.0.2
Image File               : TL_10_02_0001.swi
Configuration File       : config_file
ZTP Status               : Failed - Could not copy to start-up configuration
Aruba Central Location   : secure.arubanetworks.com
Force-Provision          : Disabled
HTTP Proxy Location      : http.proxy.arubanetworks.com
```

In the following example, the ZTP operation succeeded, and both an image file and a configuration file were provided.

```
VSF-10-Mbr# show ztp information
TFTP Server             : 10.1.84.160
Image File              : FL_10_06_0001CK.swi
Configuration File      : 102720-new-setup-config-updated.txt
Status                  : Success
Aruba Central Location : NA
Force-Provision         : Disabled
HTTP Proxy Location     : NA
VSF-10-Mbr#
```

In the following example, the ZTP option succeeded. A configuration file was not provided, but an image file was provided.

```
VSF-10-Mbr# show ztp information
TFTP Server             : 10.1.84.160
Image File              : TL_10_02_0001.swi
Configuration File      : NA
Status                  : Success
Aruba Central Location : NA
Force-Provision         : Disabled
HTTP Proxy Location     : NA
VSF-10-Mbr#
```

In the following example, the ZTP operation failed because the TFTP server was unreachable.

```
VSF-10-Mbr# show ztp information
TFTP Server             : 10.1.84.160
Image File              : TL_10_02_0001.swi
Configuration File      : 102720-new-setup-config-updated.txt
Status                  : Failed - TFTP server unreachable
Aruba Central Location : NA
Force-Provision         : Disabled
HTTP Proxy Location     : NA
VSF-10-Mbr#
```

In the following example, the ZTP operation was stopped because the switch did not receive any options from the DHCP server for ZTP within 1 minute of receiving the IP address from the server.

```
VSF-10-Mbr## show ztp information
TFTP Server             : NA
Image File              : NA
```

```
Configuration File     : NA
Status                 : Failed - Timed out while waiting to receive ZTP options
Aruba Central Location : NA
Force-Provision        : Disabled
HTTP Proxy Location    : NA
VSF-10-Mbr#
```

In the following example, the ZTP operation was stopped because the switch was booted from a configuration that was not the factory default configuration.

```
switch# show ztp information
TFTP Server            : 10.0.0.2
Image File             : TL_10_02_0001.swi
Configuration File     : ztp.cfg
Status                 : Failed - Custom startup configuration detected
Aruba Central Location : NA
Force-Provision        : Disabled
HTTP Proxy Location    : NA
```

ztp force provision

Syntax

```
ztp force-provision
no ztp force-provision
```

Description

Starts on-demand ZTP.

Command context

Operator (>) or Manager (#)

Authority

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

Usage

DHCP options received are processed independent of he current state of configuration on the switch. Previous ZTP TFTP Server, Image File, Configuration File, Aruba Central Location, and HTTP Proxy location options are cleared and the switch sends a DHCP request.

Examples

In the following example, force-provision is enabled.

```
switch# configure terminal
switch(config)# ztp force-provision
```

In the following example, force-provision status is checked while enabled.

```
switch# show ztp information
TFTP Server            : 10.0.0.2
Image File             : TL_10_02_0001.swi
```

```
Configuration File     : ztp.cfg
Status                 : Success
Aruba Central Location : NA
Force-Provision        : Enabled
HTTP Proxy Location    : NA
```

In the following example, force-provision is disabled.

```
switch# configure terminal
switch(config)# no ztp force-provision
```

In the following example, force-provision status is checked while disabled.

```
switch# show ztp information
TFTP Server            : 10.0.0.2
Image File             : TL_10_02_0001.swi
Configuration File     : ztp.cfg
Status                 : Success
Aruba Central Location : NA
Force-Provision        : Disabled
HTTP Proxy Location    : NA
```

bluetooth disable

Syntax

```
bluetooth disable

no bluetooth disable
```

Description

Disables the Bluetooth feature on the switch. The Bluetooth feature includes both Bluetooth Classic and Bluetooth Low Energy (BLE). Bluetooth is enabled by default.

The `no` form of this command enables the Bluetooth feature on the switch.

Command context

`config`

Authority

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

Example

Disabling Bluetooth on the switch. *<XXXX>* is the switch platform and *<NNNNNNNNNN>* is the device identifier.

```
switch(config)# bluetooth disable
switch# show bluetooth
Enabled              : No
Device name          : <XXXX>-<NNNNNNNNNN>

switch(config)# show running-config
...
bluetooth disabled
...
```

bluetooth enable

Syntax

```
bluetooth enable

no bluetooth enable
```

Description

This command enables the Bluetooth feature on the switch. The Bluetooth feature includes both Bluetooth Classic and Bluetooth Low Energy (BLE).

Default: Bluetooth is enabled by default.

The `no` form of this command disables the Bluetooth feature on the switch.

Command context

`config`

Authority

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

Usage

The default configuration of the Bluetooth feature is `enabled`. The output of the `show running-config` command includes Bluetooth information only if the Bluetooth feature is disabled.

The Bluetooth feature includes both Bluetooth Classic and Bluetooth Low Energy (BLE).

The Bluetooth feature requires the USB feature to be enabled. If the USB feature has been disabled, you must enable the USB feature before you can enable the Bluetooth feature.

Examples

```
switch(config)# bluetooth enable
```

clear events

Syntax

`clear events`

Description

Clears up event logs. Using the `show events` command will only display the logs generated after the `clear events` command.

Command context

Manager (`#`)

Authority

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

Examples

Clearing all generated event logs:

```
switch# show events
-------------------------------------------------
show event logs
-------------------------------------------------
2018-10-14:06:57:53.534384|hpe-sysmond|6301|LOG_INFO|MSTR|1|System resource
utilization poll interval is changed to 27
2018-10-14:06:58:30.805504|lldpd|103|LOG_INFO|MSTR|1|Configured LLDP tx-timer to 36
2018-10-14:07:01:01.577564|hpe-sysmond|6301|LOG_INFO|MSTR|1|System resource
utilization poll interval is changed to 49

switch# clear events
```

```
switch# show events
-------------------------------------------------
show event logs
-------------------------------------------------
2018-10-14:07:03:05.637544|hpe-sysmond|6301|LOG_INFO|MSTR|1|System resource
utilization poll interval is changed to 34
```

clear ip errors

Syntax

```
clear ip errors
```

Description

Clears all IP error statistics.

Command context

Manager (#)

Authority

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

Example

Clearing and showing ip errors:

```
switch# clear ip errors
switch# show ip errors
---------------------------------
Drop reason                Packets
---------------------------------
Malformed packets                0
IP address errors                0
...
```

domain-name

Syntax

```
domain-name <NAME>
no domain-name [<NAME>]
```

Description

Specifies the domain name of the switch.

The no form of this command sets the domain name to the default, which is no domain name.

Command context

```
config
```

Parameters

<NAME>

Specifies the domain name to be assigned to the switch. The first character of the name must be a letter or a number. Length: 1 to 32 characters.

Authority

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

Examples

Setting and showing the domain name:

```
switch# show domain-name

switch# config
switch(config)# domain-name example.com
switch(config)# show domain-name
example.com
switch(config)#
```

Setting the domain name to the default value:

```
switch(config)# no domain-name
switch(config)# show domain-name

switch(config)#
```

hostname

Syntax

`hostname <HOSTNAME>`

`no hostname [<HOSTNAME>]`

Description

Sets the host name of the switch.

The `no` form of this command sets the host name to the default value, which is `switch`.

Command context

`config`

Parameters

`<HOSTNAME>`

Specifies the host name. The first character of the host name must be a letter or a number. Length: 1 to 32 characters. Default: `switch`

Authority

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

Examples

Setting and showing the host name:

```
switch# show hostname
switch
switch# config
switch(config)# hostname myswitch
myswitch(config)# show hostname
myswitch
```

Setting the host name to the default value:

```
myswitch(config)# no hostname
switch(config)#
```

led locator

Syntax

```
led locator {on | off | flashing}
```

Description

Sets the state of the locator LED to `on`, `off` (default), or `flashing`.

Command context

Manager (`#`)

Parameters

`on`
: Turns on the LED.

`off`
: Turns off the LED, which is the default value.

`flashing`
: Sets the LED to blink on and off repeatedly.

Authority

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

Example

Setting the state of the locator LED:

```
switch# led locator flashing
```

module admin-state

Not supported on the 6300 Switch Series.

Syntax

```
module <SLOT-ID> admin-state {diagnostic | down | up}
```

Description

Sets the administrative state of the specified line module.

Command context

`config`

Parameters

<SLOT-ID>

Specifies the member and slot of the module. For example, to specify the module in member 1, slot 3, enter the following:

`1/3`

`admin-state {diagnostic | down | up}`

Selects the administrative state in which to put the specified module:

`diagnostic`

Selects the `diagnostic` administrative state. Network traffic does not pass through the module.

`down`

Selects the `down` administrative state. Network traffic does not pass through the module.

`up`

Selects the `up` administrative state. The line module is fully operational. The `up` state is the default administrative state.

Authority

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

Example

Setting the administrative state of the module in slot **1/3** to `down`:

```
switch(config)# module 1/3 admin-state down
```

module product-number

Not supported on the 6300 Switch Series.

Syntax

`module <SLOT-ID> product-number [<PRODUCT-NUM>]no module <SLOT-ID>`

Description

Changes the configuration of the switch to indicate that the specified member and slot number contains, or will contain, a line module.

The `no` form of this command removes the line module and its interfaces from the configuration. If there is a line module installed in the slot, the line module is powered off and then powered on.

Command context

`config`

Parameters

<SLOT-ID>

Specifies the member and slot in the form `m/s`, where `m` is the member number, and `s` is the slot number.

<PRODUCT-NUM>

Specifies the product number of the line module. For example: `JL363A`

If there is a line module installed in the slot when you execute this command, *<PRODUCT-NUM>* is optional. The switch reads the product number information from the module that is installed in the slot.

If there is no line module installed in the slot when you execute this command, *<PRODUCT-NUM>* is required.

Authority

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

Usage

The default configuration associated with a line module slot is:

- There is no module product number or interface configuration information associated with the slot. The slot is available for the installation with any supported line module.
- The Admin State is Up (which is the default value for Admin State).

To add a line module to the configuration, you must use the `module` command either before or after you install the physical module.

If you execute the `module` command after you install a line module in an empty slot, you can omit the *<PRODUCT-NUM>* variable. The switch reads the product information from the installed module.

If the module is not installed in the slot when you execute the module command, you must specify a value for the *<PRODUCT-NUM>* variable:

- The switch validates the product number of the module against the slot number you specify to ensure that the right type of module is configured for the specified slot.

For example, the switch returns an error if you specify the product number of a line module for a slot reserved for management modules.

- You can configure the line module interfaces before the line module is installed.

When you install the physical line module in a preconfigured slot, the following actions occur:

- If a product number was specified in the command and it matches the product number of the installed module, the switch initializes the module.
- If a product number was specified in the command and the product number of the module does not match what was specified, the module device initialization fails.

The `no` form of the command removes the line module and its interfaces from the configuration and restores the line module slot to the default configuration.

If there is a line module installed in the slot when you execute the `no` form of the command, the command also powers off and then powers on the module. Traffic passing through the line module is stopped. Management sessions connected through the line module are also affected.

If the slot associated with the line module is in the default configuration, you can remove the module from the chassis without disrupting the operation of the switch.

Examples

Configuring slot 1/1 for future installation of a line module:

```
switch(config)# module 1/1 product-number jl363a
```

Configuring a line module that is already installed in slot 1/1:

```
switch(config)# module 1/1 product-number
```

Attempting to configure slot 1/1 for the future installation of a line module without specifying the product number (returned error shown):

```
switch(config)# module 1/1 product-number
Line module '1/4' is not physically available.  Please provide the product
number to preconfigure the line module.
```

Removing a module from the configuration:

```
switch(config)# no module 1/1
This command will power cycle the specified line module and restore its default
configuration. Any traffic passing through the line module will be interrupted.
Management sessions connected through the line module will be affected. It
might take a few minutes to complete this operation.

Do you want to continue (y/n)? y
switch(config)#
```

mtrace

Syntax

```
mtrace <IPV4-SRC-ADDR> <IPV4-GROUP-ADDR> [lhr <IPV4-LHR-ADDR>] [ttl <HOPS>]
   [vrf <VRF-NAME>]
```

Description

Traces the specified IPv4 source and group addresses.

Command context

Manager (#)

Parameters

IPV4-SRC-ADDR

Specifies the source IPv4 address to trace.

IPV4-GROUP-ADDR

Specifies the group IPv4 address to trace.

`lhr` *<IPV4-LHR-ADDR>*

Specifies the last hop router address from which to start the trace.

`ttl` *<HOPS>*

Specifies the Time-To-Live duration in hops. Range: 1 to 255 hops. Default: 8 hops.

`vrf` *<VRF-NAME>*

Specifies the name of the VRF. If a name is not specified the default VRF will be used.

Authority

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

Examples

Tracing with source, group, and LHR addresses and TTL:

```
(switch)# mtrace 20.0.0.1 239.1.1.1 lhr 10.1.1.1 ttl 10

Type escape sequence to abort.
Mtrace from 10.0.0.1 for Source 20.0.0.1 via Group 239.1.1.1
From destination(?) to source (?)...
Querying ful reverse path...
0  10.0.0.1
-1  30.0.0.1 PIM  0 ms
-2  40.0.0.1 PIM  2 ms
-3  50.0.0.1 PIM  100 ms
-4  60.0.0.1 PIM  156 ms
-5  20.0.0.1 PIM  123 ms
```

Tracing with source and group addresses:

```
(switch)# mtrace 200.0.0.1 239.1.1.1

Type escape sequence to abort.
Mtrace from self for Source 200.0.0.1 via Group 239.1.1.1
From destination(?) to source (?)...
Querying ful reverse path...
0  10.0.0.1
-1  30.0.0.1 PIM  0 ms
-2  40.0.0.1 PIM  2 ms
-3  50.0.0.1 PIM  100 ms
-4  60.0.0.1 PIM  156 ms
-5  200.0.0.1 PIM  123 ms
```

show bluetooth

Syntax

```
show bluetooth
```

Description

Shows general status information about the Bluetooth wireless management feature on the switch.

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.

Usage

This command shows status information about the following:

- The USB Bluetooth adapter
- Clients connected using Bluetooth
- The switch Bluetooth feature.

The output of the `show running-config` command includes Bluetooth information only if the Bluetooth feature is disabled.

The device name given to the switch includes the switch serial number to uniquely identify the switch while pairing with a mobile device.

The management IP address is a private network address created for managing the switch through a Bluetooth connection.

Examples

Example output when Bluetooth is enabled but no Bluetooth adapter is connected. *<XXXX>* is the switch platform and *<NNNNNNNNNN>* is the device identifier.

```
switch# show bluetooth
Enabled              : Yes
Device name          : <XXXX>-<NNNNNNNNNN>
Adapter State        : Absent
```

Example output when Bluetooth is enabled and there is a Bluetooth adapter connected:

```
switch# show bluetooth
Enabled              : Yes
Device name          : <XXXX>-<NNNNNNNNNN>
Adapter State        : Ready
Adapter IP address   : 192.168.99.1
Adapter MAC address  : 480fcf-af153a

Connected Clients
-----------------
Name             MAC Address      IP Address     Connected Since
--------------   --------------   ------------   ------------------------
Mark's iPhone    089734-b12000    192.168.99.10  2018-07-09 08:47:22 PDT
```

Example output when Bluetooth is disabled:

```
switch# show bluetooth
Enabled              : No
Device name          : <XXXX>-<NNNNNNNNNN>
```

show boot-history

Syntax

```
show boot-history [all]
```

Description

Shows boot information. When no parameters are specified, shows the most recent information about the boot operation, and the three previous boot operations for the active management module. When the `all` parameter is specified, shows the boot information for the active management module and all available line modules. To view boot-history on the standby, the command must be sent on the standby console.

Command context

Manager (`#`)

Parameters

`all`

Shows boot information for the active management module and all available line modules.

Authority

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

Usage

This command displays the boot-index, boot-ID, and up time in seconds for the current boot. If there is a previous boot, it displays boot-index, boot-ID, reboot time (based on the time zone configured in the system) and reboot reasons. Previous boot information is displayed in reverse chronological order.

`Index`

The position of the boot in the history file. Range: 0 to3.

`Boot ID`

A unique ID for the boot . A system-generated 128-bit string.

`Current Boot, up for <SECONDS> seconds`

For the current boot, the `show boot-history` command shows the number of seconds the module has been running on the current software.

`Timestamp boot reason`

For previous boot operations, the `show boot-history` command shows the time at which the operation occurred and the reason for the boot. The reason for the boot is one of the following values:

`<DAEMON-NAME>` `crash`

The daemon identified by *<DAEMON-NAME>* caused the module to boot.

`Kernel crash`

The operating system software associated with the module caused the module to boot.

`Reboot requested through database`

The reboot occurred because of a request made through the CLI or other API.

`Uncontrolled reboot`

The reason for the reboot is not known.

Examples

Showing the boot history of the active management module:

```
switch# show boot-history
Management module
=================

Index : 3
Boot ID : f1bf071bdd04492bbf8439c6e479d612
Current Boot, up for 22 hrs 12 mins 22 secs

Index : 2
Boot ID : edfa2d6598d24e989668306c4a56a06d
07 Aug 18 16:28:01 : Reboot requested through database

Index : 1
Boot ID : 0bda8d0361df4a7e8e3acdc1dba5caad
07 Aug 18 14:08:46 : Reboot requested through database

Index : 0
Boot ID : 23da2b0e26d048d7b3f4b6721b69c110
07 Aug 18 13:00:46 : Reboot requested through database
switch#
```

Showing the boot history of the active management module and all line modules:

```
switch# show boot-history all
Management module
=================

Index : 3
Boot ID : f1bf071bdd04492bbf8439c6e479d612
Current Boot, up for 22 hrs 12 mins 22 secs

Index : 2
Boot ID : edfa2d6598d24e989668306c4a56a06d
07 Aug 18 16:28:01 : Reboot requested through database

Index : 1
Boot ID : 0bda8d0361df4a7e8e3acdc1dba5caad
07 Aug 18 14:08:46 : Reboot requested through database

Index : 0
Boot ID : 23da2b0e26d048d7b3f4b6721b69c110
07 Aug 18 13:00:46 : Reboot requested through database

Line module 1/1
=================
Index : 3
10 Aug 17 12:45:46 : dune_agent crashed
...
```

show capacities

Syntax

`show capacities <FEATURE> [vsx-peer]`

Description

Shows system capacities and their values for all features or a specific feature.

Command context

Manager (`#`)

Parameters

`<FEATURE>`

Specifies a feature. For example, `aaa` or `vrrp`.

`[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

Capacities are expressed in user-understandable terms. Thus they may not map to a specific hardware or software resource or component. They are not intended to define a feature exhaustively.

Examples

Showing all available capacities for BGP:

```
switch# show capacities bgp

System Capacities: Filter BGP
Capacities Name                                                                    Value
-----------------------------------------------------------------------------------------
Maximum number of AS numbers in as-path attribute                                     32
...
```

Showing all available capacities for mirroring:

```
switch# show capacities mirroring

System Capacities: Filter Mirroring
Capacities Name                                                                    Value
-----------------------------------------------------------------------------------------
Maximum number of Mirror Sessions configurable in a system                             4
Maximum number of enabled Mirror Sessions in a system                                  4
```

Showing all available capacities for MSTP:

```
switch# show capacities mstp

System Capacities: Filter MSTP
Capacities Name                                                                    Value
-----------------------------------------------------------------------------------------
Maximum number of mstp instances configurable in a system                             64
```

Showing all available capacities for VLAN count:

```
switch# show capacities vlan-count

System Capacities: Filter VLAN Count
Capacities Name                                                                    Value
-----------------------------------------------------------------------------------------
Maximum number of VLANs supported in the system                                     4094
```

show capacities-status

Syntax

```
show capacities-status <FEATURE> [vsx-peer]
```

Description

Shows system capacities status and their values for all features or a specific feature.

Command context

Manager (#)

Parameters

<FEATURE>

Specifies the feature, for example `aaa` or `vrrp` for which to display capacities, values, and status. Required.

`[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

Showing the system capacities status for all features:

```
switch# show capacities-status

System Capacities Status
Capacities Status Name                                                    Value Maximum
------------------------------------------------------------------------------------
Number of active gateway mac addresses in a system                        0      16
Number of aspath-lists configured                                         0      64
Number of community-lists configured                                      0      64
...
```

Showing the system capacities status for BGP:

```
switch# show capacities-status bgp

System Capacities Status: Filter BGP
Capacities Status Name                                                     Value Maximum
-------------------------------------------------------------------------------------
Number of aspath-lists configured                                          0      64
Number of community-lists configured                                       0      64
Number of neighbors configured across all VRFs                             0      50
Number of peer groups configured across all VRFs                           0      25
Number of prefix-lists configured                                          0      64
Number of route-maps configured                                            0      64
Number of routes in BGP RIB                                                0  256000
Number of route reflector clients configured across all VRFs               0      16
```

show core-dump

Syntax

`show core-dump [all | <SLOT-ID>]`

Description

Shows core dump information about the specified module. When no parameters are specified, shows only the core dumps generated in the current boot of the management module. When the `all` parameter is specified, shows all available core dumps.

Command context

Manager (`#`)

Parameters

`all`

Shows all available core dumps.

`<SLOT-ID>`

Shows the core dumps for the management module or line module in *`<SLOT-ID>`*. *`<SLOT-ID>`* specifies a physical location on the switch. Use the format `member/slot/port` (for example, `1/3/1`) for line modules. Use the format `member/slot` for management modules.

You must specify the slot ID for either the active management module, or the line module.

Authority

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

Usage

When no parameters are specified, the `show core-dump` command shows only the core dumps generated in the current boot of the management module. You can use this command to determine when any crashes are occurring in the current boot.

If no core dumps have occurred, the following message is displayed: `No core dumps are present`

To show core dump information for the standby management module, you must use the `standby` command to switch to the standby management module and then execute the `show core-dump` command.

In the output, the meaning of the information is the following:

`Daemon Name`

Identifies name of the daemon for which there is dump information.

`Instance ID`

Identifies the specific instance of the daemon shown in the `Daemon Name` column.

`Present`

Indicates the status of the core dump:

`Yes`

The core dump has completed and available for copying.

`In Progress`

Core dump generation is in progress. Do not attempt to copy this core dump.

`Timestamp`

Indicates the time the daemon crash occurred. The time is the local time using the time zone configured on the switch.

`Build ID`

Identifies additional information about the software image associated with the daemon.

Examples

Showing core dump information for the current boot of the active management module only:

```
switch# show core-dump
================================================================================
Daemon Name     | Instance ID | Present    | Timestamp             | Build ID
================================================================================
hpe-fand          1399          Yes          2017-08-04 19:05:34      1246d2a
hpe-sysmond       957           Yes          2017-08-04 19:05:29      1246d2a
================================================================================
Total number of core dumps : 2
================================================================================
```

Showing all core dumps:

```
switch# show core-dump all
================================================================================
Management Module core-dumps
================================================================================
Daemon Name     | Instance ID | Present    | Timestamp           | Build ID
================================================================================
hpe-sysmond       513           Yes          2017-07-31 13:58:05    e70f101
hpe-tempd         1048          Yes          2017-08-13 13:31:53    e70f101
hpe-tempd         1052          Yes          2017-08-13 13:41:44    e70f101

Line Module core-dumps
================================================================================
Line Module : 1/1
================================================================================
dune_agent_0      18958         Yes          2017-08-12 11:50:17    e70f101
dune_agent_0      18842         Yes          2017-08-12 11:50:09    e70f101
================================================================================
Total number of core dumps : 5
================================================================================
```

show domain-name

Syntax

`show domain-name` [vsx-peer]

Description

Shows the current domain name.

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.

Usage

If there is no domain name configured, the CLI displays a blank line.

Example

Setting and showing the domain name:

```
switch# show domain-name

switch# config
switch(config)# domain-name example.com
switch(config)# show domain-name
```

```
example.com
switch(config)#
```

show environment fan

Syntax

```
show environment fan [vsf | vsx-peer]
```

Description

Shows the status information for all fans and fan trays (if present) in the system.

Command context

Manager (#)

Parameters

`vsf`

Shows output from the VSF member-id on switches that support VSF.

`[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

For fan trays, `Status` is one of the following values:

`ready`

The fan tray is operating normally.

`fault`

The fan tray is in a fault event. The status of the fan tray does not indicate the status of fans.

`empty`

The fan tray is not installed in the system.

For fans:

`Speed`

Indicates the relative speed of the fan based on the nominal speed range of the fan. Values are:

`Slow`

The fan is running at less than 25% of its maximum speed.

`Normal`

The fan is running at 25-49% of its maximum speed.

`Medium`

The fan is running at 50-74% of its maximum speed.

`Fast`

The fan is running at 75-99% of its maximum speed.

`Max`

The fan is running at 100% of its maximum speed.

`N/A`

The fan is not installed.

`Direction`

The direction of airflow through the fan. Values are:

`front-to-back`

Air flows from the front of the system to the back of the system.

`N/A`

The fan is not installed.

`Status`

Fan status. Values are:

`uninitialized`

The fan has not completed initialization.

`ok`

The fan is operating normally.

`fault`

The fan is in a fault state.

`empty`

The fan is not installed.

Examples

Showing output for systems with fan trays for 6400 switch series:

```
switch# show environment fan
Fan tray information
-------------------------------------------------------------------------------
Mbr/Tray  Description                              Status  Serial Number  Fans
-------------------------------------------------------------------------------
1/1       R0X32A Aruba 6400 Fan Tray               ready   SG9ZKJL7JW     4
1/2       R0X32A Aruba 6400 Fan Tray               ready   SG9ZKJL7GL     4
1/3       R0X32A Aruba 6400 Fan Tray               ready   SG9ZKJL78L     4
1/4       R0X32A Aruba 6400 Fan Tray               ready   SG9ZKJL7GJ     4
Fan information
-------------------------------------------------------------------------------
Mbr/Tray/Fan  Product  Serial Number  Speed   Direction      Status  RPM
              Name
-------------------------------------------------------------------------------
1/1/1         N/A      N/A            slow    front-to-back  ok      5371
1/1/2         N/A      N/A            slow    front-to-back  ok      5320
1/1/3         N/A      N/A            slow    front-to-back  ok      5328
1/1/4         N/A      N/A            slow    front-to-back  ok      5256
1/2/1         N/A      N/A            slow    front-to-back  ok      5371
1/2/2         N/A      N/A            slow    front-to-back  ok      5349
1/2/3         N/A      N/A            slow    front-to-back  ok      5292
1/2/4         N/A      N/A            slow    front-to-back  ok      5349
1/3/1         N/A      N/A            slow    front-to-back  ok      5313
1/3/2         N/A      N/A            slow    front-to-back  ok      5371
1/3/3         N/A      N/A            slow    front-to-back  ok      5379
1/3/4         N/A      N/A            slow    front-to-back  ok      5379
1/4/1         N/A      N/A            slow    front-to-back  ok      5313
1/4/2         N/A      N/A            slow    front-to-back  ok      5299
1/4/3         N/A      N/A            slow    front-to-back  ok      5285
1/4/4         N/A      N/A            slow    front-to-back  ok      5371
```

Showing output for a system without a fan tray:

```
switch# show environment fan

Fan information
-------------------------------------------------------------------
Fan   Serial Number   Speed   Direction      Status        RPM
-------------------------------------------------------------------
1     SGXXXXXXXXXX    slow    front-to-back  ok            6000
2     SGXXXXXXXXXX    normal  front-to-back  ok            8000
3     SGXXXXXXXXXX    medium  front-to-back  ok            11000
4     SGXXXXXXXXXX    fast    front-to-back  ok            14000
5     SGXXXXXXXXXX    max     front-to-back  fault         16500
6     N/A             N/A     N/A            empty
...
```

show environment led

Syntax

```
show environment led <MEMBER-ID> [vsx-peer]
```

Description

Shows state and status information for all the configurable LEDs in the system.

Command context

Operator (>) or Manager (#)

Parameters

<MEMBER-ID>

Shows output from the specified VSF member ID on switches that support VSF.

[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 state and status for LED:

```
switch# show environment led
Mbr/Name       State     Status
--------------------------------
1/locator      off       ok
```

show environment power-consumption

Not supported on the 6300 Switch Series.

Syntax

```
show environment power-consumption [vsx-peer]
```

Description

Shows the power being consumed by each management module, line card, and fabric card subsystem, and shows power consumption for the entire chassis.

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.

Usage

This command is only applicable to systems that support power consumption readings.

The power consumption values are updated once every minute.

The output of this command includes the following information:

`Name`

Shows the member number and slot number of the management module, line module, or fabric card module.

`Type`

Shows the type of module installed at the location specified by `Name`.

`Description`

Shows the product name and brief description of the module.

`Usage`

Shows the instantaneous power consumption of the module. Power consumption is shown in Watts.

`Module Total Power Usage`

Shows the total power consumption of all the modules listed. Power consumption is shown in Watts.

`Chassis Total Power Usage`

Shows the total instantaneous power consumed by the entire chassis, including modules and components that do not support individual power reporting. Power consumption is shown in Watts.

`Chassis Total Power Available`

Shows the total amount of power, in Watts, that can be supplied to the chassis.

`Chassis Total Power Allocated`

Shows total power, in Watts, that is allocated to powering the chassis and its installed modules.

`Chassis Total Power Unallocated`

Shows the total amount of power, in Watts, that has not been allocated to powering the chassis or its installed modules. This power can be used for additional hardware you install in the chassis.

Example

Showing the power consumption for an Aruba 6400 switch:

```
switch> show environment power-consumption
                                                                    Power
Name    Type                Description                             Usage
```

```
--------------------------------------------------------------------------------
1/1     management-module   R0X31A 6400 Management Module              18 W
1/2     management-module                                              0 W
1/3     line-card-module                                               0 W
1/4     line-card-module    R0X39A 6400 48p 1GbE CL4 PoE 4SFP56 Mod    54 W
1/5     line-card-module                                               0 W
1/6     line-card-module    R0X39A 6400 48p 1GbE CL4 PoE 4SFP56 Mod    56 W
1/7     line-card-module    R0X39A 6400 48p 1GbE CL4 PoE 4SFP56 Mod    51 W
1/1     fabric-card-module  R0X24A 6405 Chassis                        71 W


Module Total Power Usage                                               250 W
Chassis Total Power Usage                                              294 W

Chassis Total Power Available                                          1800 W
```

show environment power-supply

Syntax

```
show environment power-supply [vsf | vsx-peer]
```

Description

Shows status information about all power supplies in the switch.

Command context

Operator (>) or Manager (#)

Parameters

`vsf`

Shows output from the VSF member-id on switches that support VSF.

`[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 following information is provided for each power supply:

`Mbr/PSU`

Shows the member and slot number of the power supply.

`Product Number`

Shows the product number of the power supply.

`Serial Number`

Shows the serial number of the power supply, which uniquely identifies the power supply.

`PSU Status`

The status of the power supply. Values are:

`OK`

Power supply is operating normally.

`OK*`

Power supply is operating normally, but it is the only power supply in the chassis. One power supply is not sufficient to supply full power to the switch. When this value is shown, the output of the command also shows a message at the end of the displayed data.

`Absent`

No power supply is installed in the specified slot.

`Input fault`

The power supply has a fault condition on its input.

`Output fault`

The power supply has a fault condition on its output.

`Warning`

The power supply is not operating normally.

`Wattage Maximum`

Shows the maximum amount of wattage that the power supply can provide.

Example

Showing the output when only one power supply is installed in an Aruba 6400 switch chassis:

```
switch#  show environment power-supply
         Product  Serial                PSU           Wattage
Mbr/PSU  Number   Number                Status        Maximum
-------------------------------------------------------------
1/1      R0X36A   CN91KMM2H3            OK            3000
1/2      N/A      N/A                   Absent        0
1/3      N/A      N/A                   Absent        0
1/4      N/A      N/A                   Absent        0
```

show environment rear-display-module

Syntax

```
show environment rear-display-module [vsx-peer]
```

Description

Shows information about the display module on the back of the switch.

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

Showing the rear display module information on the back of the switch:

```
switch> show environment rear-display-module

Rear display module is ready
Description: 8400 Rear Display Mod
Full Description: 8400 Rear Display Module
Serial number: SG00000000
Part number: 5300_0272
```

show environment temperature

Syntax

```
show environment temperature [detail] [vsf | vsx-peer]
```

Description

Shows the temperature information from sensors in the switch that affect fan control.

Command context

Operator (>) or Manager (#)

Parameters

`detail`

Shows detailed information from each temperature sensor.

`vsf`

Shows output from the VSF member-id on switches that support VSF

`[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

Temperatures are shown in Celsius.

Valid values for status are the following:

`normal`

Sensor is within nominal temperature range.

`min`

Lowest temperature from this sensor.

`max`

Highest temperature from this sensor.

`low_critical`

Lowest threshold temperature for this sensor.

`critical`

Highest threshold temperature for this sensor.

`fault`

Fault event for this sensor.

`emergency`

Over temperature event for this sensor.

Examples

Showing current temperature information for a 6300 switch:

```
switch# show environment temperature
Temperature information
-------------------------------------------------------------------------------
                                                      Current
Mbr/Slot-Sensor                 Module Type        temperature  Status
-------------------------------------------------------------------------------
1/1-PHY-01-04                   line-card-module     45.00 C    normal
1/1-PHY-05-08                   line-card-module     45.00 C    normal
1/1-PHY-09-12                   line-card-module     46.00 C    normal
1/1-PHY-13-16                   line-card-module     47.00 C    normal
1/1-PHY-17-20                   line-card-module     47.00 C    normal
1/1-PHY-21-24                   line-card-module     50.00 C    normal
1/1-PHY-25-28                   line-card-module     45.00 C    normal
1/1-PHY-29-32                   line-card-module     47.00 C    normal
1/1-PHY-33-36                   line-card-module     48.00 C    normal
1/1-PHY-37-40                   line-card-module     47.00 C    normal
1/1-PHY-41-44                   line-card-module     48.00 C    normal
1/1-PHY-45-48                   line-card-module     49.00 C    normal
1/1-Switch-ASIC-Internal        line-card-module     56.25 C    normal

1/1-CPU-Zone-0                  management-module    50.00 C    normal
1/1-CPU-Zone-1                  management-module    50.00 C    normal
1/1-CPU-Zone-2                  management-module    50.00 C    normal
1/1-CPU-Zone-3                  management-module    51.00 C    normal
1/1-CPU-Zone-4                  management-module    51.00 C    normal
1/1-CPU-diode                   management-module    53.12 C    normal
1/1-DDR                         management-module    45.25 C    normal
1/1-Inlet-Air                   management-module    24.88 C    normal
1/1-MB-IBC                      management-module    45.62 C    normal
1/1-Switch-ASIC-diode           management-module    58.06 C    normal
```

Showing detailed temperature information for a 6300 switch:

```
switch# show environment temperature detail
Detailed temperature information
-------------------------------------------------------------------
Mbr/Slot-Sensor      : 1/1-PHY-01-04
Module Type          : line-card-module
Module Description   : JL659A 6300M 48SR5 CL6 PoE 4SFP56 Swch
Status               : normal
Fan-state            : normal
Current temperature  : 45.00 C
Minimum temperature  : 41.00 C
Maximum temperature  : 50.00 C

Mbr/Slot-Sensor      : 1/1-PHY-05-08
Module Type          : line-card-module
Module Description   : JL659A 6300M 48SR5 CL6 PoE 4SFP56 Swch
Status               : normal
Fan-state            : normal
Current temperature  : 45.00 C
Minimum temperature  : 41.00 C
Maximum temperature  : 50.00 C

...
```

show events

Syntax

```
show events [ -e <EVENT-ID> |
    -s {alert | crit | debug | emer | err | info | notice | warn} |
    -r | -a | -i  <MEMBER/SLOT> | -n <count> |
    -m {active | standby} |
    -c {lldp | ospf | ... | } |
    -d {lldpd | hpe-fand | ... |}]
```

For 6300 switches:

```
show events [ -e <EVENT-ID> |
     -s {alert | crit | debug | emer | err | info | notice | warn} |
     -r | -a | -i <count> |
     -m {master | standby} |
     -c {lldp | ospf | ... | } |
     -d {lldpd | hpe-fand | ... |}]
```

Description

Shows event logs generated by the switch modules since the last reboot.

Command context

Manager (#)

Parameters

`-e <EVENT-ID>`

Shows the event logs for the specified event ID. Event ID range: 101 through 99999.

`-s {alert | crit | debug | emer | err | info | notice | warn}`

Shows the event logs for the specified severity. Select the severity from the following list:

- `alert`: Displays event logs with severity alert and above.
- `crit`: Displays event logs with severity critical and above.
- `debug`: Displays event logs with all severities.
- `emer`: Displays event logs with severity emergency only.
- `err`: Displays event logs with severity error and above.
- `info`: Displays event logs with severity info and above.
- `notice`: Displays event logs with severity notice and above.
- `warn`: Displays event logs with severity warning and above.

`-r`

Shows the most recent event logs first.

`-a`

Shows all event logs, including those events from previous boots.

`-i <MEMBER-SLOT>`

Shows the event logs for the specified slot ID on a 6400 switch.

`-i <MEMBER-ID>`

Shows the event logs for the specified VSF member ID on a 6300 switch.

`-n <count>`

Displays the specified number of event logs.

`-m {active | standby}`

Shows the event logs for the specified management card role on an 8400 or 6400 switch. Selecting *active* displays the event log for the AMM management card role and *standby* displays event logs for the SMM management card role.

`-m {master | standby}`

Shows the event logs for the specified role on a 6200 or 6300 switch. Selecting *master* displays the event log for the VSF master role and *standby* displays event logs for the VSF standby role.

`-c {lldp | ospf | ... | }`

Shows the event logs for the specified event category. Enter `show event -c` for a full listing of supported categories with descriptions.

`-d {lldpd | hpe-fand | ... |}`

Shows the event logs for the specified process. Enter `show event -d` for a full listing of supported daemons with descriptions.

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.

Examples

Showing event logs:

```
switch# show events
-------------------------------------------------
show event logs
-------------------------------------------------
2016-12-01:12:37:31.733551|lacpd|15007|INFO|AMM|1|LACP system ID set to
70:72:cf:51:50:7c
2016-12-01:12:37:31.734541|intfd|4001|INFO|AMM|1|Interface port_admin set to up for
bridge_normal interface
2016-12-01:12:37:32.583256|switchd|24002|ERR|AMM|1|Failed to create VLAN 1 in
Hardware
```

Showing the most recent event logs first:

```
switch# show events -r
-------------------------------------------------
show event logs
-------------------------------------------------
2016-12-01:12:37:32.583256|switchd|24002|ERR|AMM|1|Failed to create VLAN 1 in
Hardware
2016-12-01:12:37:31.734541|intfd|4001|INFO|AMM|1|Interface port_admin set to up for
bridge_normal interface
2016-12-01:12:37:31.733551|lacpd|15007|INFO|AMM|1|LACP system ID set to
70:72:cf:51:50:7c
```

Showing all event logs:

```
switch# show events -a
-------------------------------------------------
show event logs
-------------------------------------------------
2016-12-01:12:37:31.733551|lacpd|15007|INFO|AMM|1|LACP system ID set to
70:72:cf:51:50:7c
2016-12-01:12:37:31.734541|intfd|4001|INFO|AMM|1|Interface port_admin set to up for
bridge_normal interface
2016-12-01:12:37:32.583256|switchd|24002|ERR|AMM|1|Failed to create VLAN 1 in
Hardware
```

Showing event logs related to the DHCP relay agent:

```
switch# show events -c dhcp-relay
2016-05-31:06:26:27.363923|hpe-relay|110001|LOG_INFO|DHCP Relay Enabled
2016-05-31:07:08:51.351755|hpe-relay|110002|LOG_INFO|DHCP Relay Disabled
```

Showing event logs related to the DHCPv6 relay agent:

```
switch# show events -c dhcpv6-relay
2016-05-31:06:26:27.363923|hpe-relay|109001|LOG_INFO|DHCPv6 Relay Enabled
2016-05-31:07:08:51.351755|hpe-relay|109002|LOG_INFO|DHCPv6 Relay Disabled
```

Showing event logs related to IRDP:

```
switch# switch# show events -c irdp
2016-05-31:06:26:27.363923|hpe-rdiscd|111001|LOG_INFO|IRDP enabled on interface %s
2016-05-31:07:08:51.351755|hpe-rdiscd|111002|LOG_INFO|IRDP disabled on interface %s
```

Showing event logs related to LACP:

```
switch# show events -c lacp
-------------------------------------------------
show event logs
-------------------------------------------------
2016-12-01:12:37:31.733551|lacpd|15007|INFO|AMM|1|LACP system ID set to
70:72:cf:51:50:7c
```

Showing event logs as per the specified management card role for a switch:

```
switch# show events -m master
-------------------------------------------------
show event logs
-------------------------------------------------
2020-04-22T05:36:13.348594+00:00 6300 lldpd[3055]: Event|109|LOG_
INFO|MSTR|1|Configured LLDP tx-delay to 2
2020-04-22T05:36:14.430166+00:00 6300 vrfmgrd[3053]: Event|5401|LOG_
INFO|MSTR|1|Created a vrf entity b1721d27-41c2-485d-9bae-2cfcbc9bd13d
2020-04-22T05:36:14.942597+00:00 6300 vrfmgrd[3053]: Event|5401|LOG_
INFO|MSTR|1|Created a vrf entity 5eb532c9-5b4d-4d83-b34a-db24ae542d4e
```

Showing event logs as per the specified slot ID:

```
switch# show events -i 1
-------------------------------------------------
Event logs from current boot
-------------------------------------------------
2020-04-22T05:36:14.430166+00:00 6300 vrfmgrd[3053]: Event|5401|LOG_
INFO|MSTR|1|Created a vrf entity b1721d27-41c2-485d-9bae-2cfcbc9bd13d
2020-04-22T05:36:14.942597+00:00 6300 vrfmgrd[3053]: Event|5401|LOG_
INFO|MSTR|1|Created a vrf entity 5eb532c9-5b4d-4d83-b34a-db24ae542d4e
2020-04-22T05:36:15.886252+00:00 6300 vsfd[710]: Event|9903|LOG_INFO|MSTR|1|Master 1
boot complete
```

Showing event logs as per the specified process:

```
switch# show events -d lacpd
-------------------------------------------------
show event logs
-------------------------------------------------
2016-12-01:12:37:31.733551|lacpd|15007|INFO|AMM|1|LACP system ID set to
70:72:cf:51:50:7c
```

Displaying the specified number of event logs:

```
switch# show events -n 5
-------------------------------------------------
show event logs
-------------------------------------------------
2018-03-21:06:12:15.500603|arpmgrd|6101|LOG_INFO|AMM|-|ARPMGRD daemon has started
2018-03-21:06:12:17.734405|lldpd|109|LOG_INFO|AMM|-|Configured LLDP tx-delay to 2
2018-03-21:06:12:17.740517|lacpd|1307|LOG_INFO|AMM|-|LACP system ID set to
70:72:cf:d4:34:42
2018-03-21:06:12:17.743491|vrfmgrd|5401|LOG_INFO|AMM|-|Created a vrf entity
42cc3df7-1113-412f-b5cb-e8227b8c22f2
2018-03-21:06:12:17.904008|vrfmgrd|5401|LOG_INFO|AMM|-|Created a vrf entity
4409133e-2071-4ab8-adfe-f9662c06b889
```

show fabric

Not supported on the 6300 Switch Series.

Syntax

`show fabric [<SLOT-ID>] [vsx-peer]`

Description

Shows information about the installed fabrics.

Command context

Operator (>) or Manager (#)

Parameters

<SLOT-ID>

Specifies the member and slot of the fabric to show. For example, to show the module in member 1, slot 2, enter the following:

`1/2`

`[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 all fabrics on Aruba 6400 switches that have two fabrics:

```
switch# show fabric

Fabric Modules
==============

     Product                                    Serial
Name Number  Description                        Number     Status
---- ------- ---------------------------------  ---------- ----------------
1/1  R0X25A  6410 Chassis                       SG9ZKM9999 Ready
1/2  R0X25A  6410 Chassis                       SG9ZKM9999 Ready
```

Showing all fabrics on Aruba 6400 switches that have one fabric:

```
switch# show fabric

Fabric Modules
==============

     Product                                    Serial
Name Number  Description                        Number     Status
---- ------- ---------------------------------- ---------- ----------------
1/1  R0X24A  6405 Chassis                       SG9ZKM9076 Ready
```

Showing a single fabric module on Aruba 6400 switches:

```
switch# show fabric 1/1

Fabric module 1/1 is ready
Admin state: Up
Description: 6405 Chassis
Full Description: 6405 Chassis
Serial number: SG00000000
Product number: R0X24A
```

show hostname

Syntax

```
show hostname [vsx-peer]
```

Description

Shows the current host name.

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.

Example

Setting and showing the host name:

```
switch# show hostname
switch
switch# config
switch(config)# hostname myswitch
myswitch(config)# show hostname
myswitch
```

show images

Syntax

```
show images [vsx-peer]
```

Description

Shows information about the software in the primary and secondary images.

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

Showing the primary and secondary images on a 6300 switch:

```
switch(config)# show images
--------------------------------------------------------------------------------
ArubaOS-CX Primary Image
--------------------------------------------------------------------------------
Version : FL.xx.xx.xxxx
Size    : 722 MB
Date    : 2019-10-22 17:00:46 PDT
SHA-256 : 4c84e49c0961fc56b5c7eab064750a333f1050212b7ce2fab587d13469d24cfa

--------------------------------------------------------------------------------
ArubaOS-CX Secondary Image
--------------------------------------------------------------------------------
Version : FL.xx.xx.xxxx
Size    : 722 MB
Date    : 2019-10-22 17:00:46 PDT
SHA-256 : 4c84e49c0961fc56b5c7eab064750a333f1050212b7ce2fab587d13469d24cfa

Default Image : secondary
```

```
-------------------------------------------------------
Management Module 1/1 (Active)
-------------------------------------------------------
Active Image       : secondary
Service OS Version : FL.01.05.0001-internal
BIOS Version       : FL.01.0001
```

Showing the primary and secondary images on a 6400 switch:

```
switch(config)# show images
-------------------------------------------------------------------------------
ArubaOS-CX Primary Image
-------------------------------------------------------------------------------
Version : FL.xx.xx.xxxxQ-2710-gd4ac39f30c9
Size    : 766 MB
Date    : 2019-10-30 17:22:01 PDT
SHA-256 : e560ca9141f425d19024d122573c5ff730df2a9a726488212263b45ea00382cf

-------------------------------------------------------------------------------
ArubaOS-CX Secondary Image
-------------------------------------------------------------------------------
Version : FL.xx.xx.xxxx
Size    : 722 MB
Date    : 2019-10-21 19:36:26 PDT
SHA-256 : 657e28adc1b512217ce780e3523c37c94db3d3420231deac1ab9aaa8324dc6b9

Default Image : secondary

-------------------------------------------------------
Management Module 1/1 (Active)
-------------------------------------------------------
Active Image       : secondary
Service OS Version : FL.01.05.0001-internal
BIOS Version       : FL.01.0001
```

show ip errors

Syntax

```
show ip errors [vsx-peer]
```

Description

Shows IP error statistics for packets received by the switch since the switch was last booted.

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.

Usage

IP error info about received packets is collected from each active line card on the switch and is preserved during failover events. Error counts are cleared when the switch is rebooted.

Drop reasons are the following:

- **Malformed packet**

  The packet does not conform to TCP/IP protocol standards such as packet length or internet header length.

  A large number of malformed packets can indicate that there are hardware malfunctions such as loose cables, network card malfunctions, or that a DOS (denial of service) attack is occurring.

- **IP address error**

  The packet has an error in the destination or source IP address. Examples of IP address errors include the following:

  - The source IP address and destination IP address are the same.
  - There is no destination IP address.
  - The source IP address is a multicast IP address.
  - The forwarding header of an IPv6 address is empty.
  - There is no source IP address for an IPv6 packet.

- **Invalid TTLs**

  The TTL (time to live) value of the packet reached zero. The packet was discarded because it traversed the maximum number of hops permitted by the TTL value.

  TTLs are used to prevent packets from being circulated on the network endlessly.

Example

Showing ip error statistics for packets received by the switch:

```
switch# show ip errors
---------------------------------
Drop reason                Packets
---------------------------------
Malformed packets                1
IP address errors               10
...
```

show module

Syntax

```
show module [<SLOT-ID>] [vsx-peer]
```

Description

Shows information about installed line modules and management modules.

Command context

Operator (>) or Manager (#)

Parameters

<SLOT-ID>

Specifies the member and slot numbers in format `member/slot`. For example, to show the module in member 1, slot 3, enter `1/3`.

`[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

Identifies and shows status information about the line modules and management modules that are installed in the switch.

If you use the *<SLOT-ID>* parameter to specify a slot that does not have a line module installed, a message similar to the following example is displayed:

```
Module 1/4 is not physically present.
```

To show the configuration information—if any—associated with that line module slot, use the `show running-configuration` command.

Status is one of the following values:

`Active`

This module is the active management module.

`Standby`

This module is the standby management module.

`Deinitializing`

The module is being deinitialized.

`Diagnostic`

The module is in a state used for troubleshooting.

`Down`

The module is physically present but is powered down.

`Empty`

The module hardware is not installed in the chassis.

`Failed`

The module has experienced an error and failed.

`Failover`

This module is a fabric module or a line module, and it is in the process of connecting to the new active management module during a management module failover event.

`Initializing`

The module is being initialized.

`Present`

The module hardware is installed in the chassis.

`Ready`

The module is available for use.

`Updating`

A firmware update is being applied to the module.

Examples

Showing all installed modules (Aruba 6300 switch):

```
switch(config)# show module

Management Modules
==================

     Product                                          Serial
Name Number  Description                              Number     Status
---- ------- ---------------------------------------- ---------- ----------------
1/1  JL659A  6300M 48SR5 CL6 PoE 4SFP56 Swch          ID9ZKHN090 Active (local)


Line Modules
============

     Product                                          Serial
Name Number  Description                              Number     Status
---- ------- ---------------------------------------- ---------- ----------------
1/1  JL659A  6300M 48SR5 CL6 PoE 4SFP56 Swch          ID9ZKHN090 Ready
```

Showing a line module (Aruba 6400 switch):

```
switch# show module 1/3

Line module 1/3 is ready
 Admin state: Up
 Description: 6400 24p 10GT 4SFP56 Mod
 Full Description: 6400 24-port 10GBASE-T and 4-port SFP56 Module
 Serial number: SG9ZKMS045
 Product number: R0X42A
 Power priority: 128
```

Showing a slot that does not contain a line module:

```
switch(config)# show module 1/3
Module 1/3 is not physically present
```

show running-config

Syntax

`show running-config [<FEATURE>] [all] [vsx-peer]`

Description

Shows the current nondefault configuration running on the switch. No user information is displayed.

Command context

Manager (#)

Parameters

<FEATURE>

Specifies the name of a feature. For a list of feature names, enter the `show running-config` command, followed by a space, followed by a question mark (?). When the `json` parameter is used, the `vsx-peer` parameter is not applicable.

all

Shows all default values for the current running configuration.

[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

Showing the current running configuration:

```
switch> show running-config
Current configuration:
!
!Version ArubaOS-CX 10.0X.XXXX
!
lldp enable
linecard-module LC1 part-number JL363A
vrf green
!
!
!
!
!
!
aaa authentication login default local
aaa authorization commands default none
!
!
!
!
vlan 1
    no shutdown
vlan 20
    no shutdown
vlan 30
    no shutdown
interface 1/1/1
    no shutdown
    no routing
    vlan access 30
interface 1/1/32
    no shutdown
    no routing
    vlan access 20
interface bridge_normal-1
    no shutdown
interface bridge_normal-2
    no shutdown
interface vlan20
    no shutdown
    vrf attach green
    ip address 20.0.0.44/24
    ip ospf 1 area 0.0.0.0
    ip pim-sparse enable

interface vlan30
    no shutdown
```

```
    vrf attach green
    ip address 30.0.0.44/24
    ip ospf 1 area 0.0.0.0
    ip pim-sparse enable

    ip pim-sparse hello-interval 100
```

Showing the current running configuration in json format:

```
switch> show running-config json
Running-configuration in JSON:
{
    "Monitoring_Policy_Script": {
        "system_resource_monitor_mm1.1.0": {
            "Monitoring_Policy_Instance": {
                "system_resource_monitor_mm1.1.0/system_resource_monitor_
mm1.1.0.default": {
                    "name": "system_resource_monitor_mm1.1.0.default",
                    "origin": "system",
                    "parameters_values": {
                        "long_term_high_threshold": "70",
                        "long_term_normal_threshold": "60",
                        "long_term_time_period": "480",
                        "medium_term_high_threshold": "80",
                        "medium_term_normal_threshold": "60",
                        "medium_term_time_period": "120",
                        "short_term_high_threshold": "90",
                        "short_term_normal_threshold": "80",
                        "short_term_time_period": "5"
                    }
                }
            },
...
...
...
...
```

Show the current running configuration without default values:

```
switch(config)# show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.04.0000-6523-gbb15c03~dirty
led locator on
!
!
!
!
!
!
!
!
!
vlan 1
switch(config)# show running-config all
Current configuration:
!
!Version ArubaOS-CX Virtual.10.04.0000-6523-gbb15c03~dirty
led locator on
```

```
!
!
!
!
!
!
!
!
!
vlan 1
switch(config)#
```

Show the current running configuration with default values:

```
switch(config)# snmp-server vrf mgmt
switch(config)# show running-config
Current configuration:
!
!Version ArubaOS-CX Virtual.10.04.0000-6523-gbb15c03~dirty
led locator on
!
!
!
!
snmp-server vrf mgmt
!
!
!
!
!
vlan 1
switch(config)#
switch(config)#
switch(config)# show running-config all
Current configuration:
!
!Version ArubaOS-CX Virtual.10.04.0000-6523-gbb15c03~dirty
led locator on
!
!
!
!
snmp-server vrf mgmt
snmp-server agent-port 161
snmp-server community public
!
!
!
!
!
vlan 1
switch(config)#
```

show running-config current-context

Syntax

```
show running-config current-context
```

Description

Shows the current non-default configuration running on the switch in the current command context.

Command context

`config` or a child of `config`. See Usage.

Authority

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

Usage

You can enter this command from the following configuration contexts:

- Any child of the global configuration (`config`) context. If the child context has instances—such as interfaces—you can enter the command in the context of a specific instance. Support for this command is provided for one level below the `config` context. For example, entering this command for a child of a child of the `config` context not supported. If you enter the command on a child of the `config` context, the current configuration of that context and the children of that context are displayed.
- The global configuration (`config`) context. If you enter this command in the global configuration (`config`) context, it shows the running configuration of the entire switch. Use the `show running-configuration` command instead.

Examples

On the 6400 Switch Series, interface identification differs.

Showing the running configuration for the current interface:

```
switch(config-if)# show running-config current-context
interface 1/1/1
vsx-sync qos vlans
    no shutdown
    description Example interface
vlan access 1
    exit
```

Showing the current running configuration for the management interface:

```
switch(config-if-mgmt)# show running-config current-context
interface mgmt
    no shutdown
    ip static 10.0.0.1/24
    default-gateway 10.0.0.8
    nameserver 10.0.0.1
```

Showing the running configuration for the external storage share named `nasfiles`:

```
switch(config-external-storage-nasfiles)# show running-config current-context
external-storage nasfiles
   address 192.168.0.1
   vrf default
   username nasuser
   password ciphertext AQBapalKj+XMsZumHEwIc9OR6YcOw5Z6Bh9rV+9ZtKDKzvbaBAAAAB1CTrM=
   type scp
   directory /home/nas
   enable
```

```
switch(config-external-storage-nasfiles)#
```

Showing the running configuration for a context that does not have instances:

```
switch(config-vsx)# show run current-context
vsx
    inter-switch-link 1/1/1
    role secondary
    vsx-sync sflow time
```

show startup-config

Syntax

```
show startup-config [json]
```

Description

Shows the contents of the startup configuration.

Switches in the `factory-default` configuration do not have a startup configuration to display.

Command context

Manager (`#`)

Parameters

`json`

Display output in JSON format.

Authority

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

Examples

Showing the startup-configuration in non-JSON format for a 6300 switch:

```
switch(config)# show startup-config
Startup configuration:
!
!Version ArubaOS-CX FL.xx.xx.xxxx
!export-password: default
hostname BLDG01-F1
user admin group administrators password ciphertext

AQBapWl8I2ZunZ43NE/8KlbQ7zYC4gTT6uSFYi6n6wyY9PdBYgAAACONCR/3+AcNvzRBch0DoG7W9z84LpJA
+6C9SKfNwCqi5/
nUPk/ZOvN91/EQXvPNkHtBtQWyYZqfkebbEH78VWRHfWZjApv4II9qmQfxpA79wEvzshdzZmuAKrm
user ateam group administrators password ciphertext

AQBapcPqMXoF+H10NKrqAedXLvlSRwf4wUEL22hXGD6ZBhicYgAAAGsbh70DKg1u+Ze1wxgmDXjkGO3bseYi
R3LKQg66vrfrqR/
M3oLlliPdZDnq9XMMvCL+7jBbYhYes8+uDxuSTh8kdkd/qj3lo5FUuC5fENgCjU0YI1l7qtU+YEnsj
!
```

```
!
!
!
radius-server host 10.10.10.15
!
radius dyn-authorization enable
ssh server vrf default
ssh server vrf mgmt
!
!
!
!
!
router ospf 1
    router-id 1.63.63.1
    area 0.0.0.0
vlan 1
vlan 66
    name vlan66
vlan 67
    name vlan67
vlan 999
    name vlan999
vlan 4000
spanning-tree
interface mgmt
    no shutdown
    ip static 10.6.9.15/24
    default-gateway 10.6.9.1
```

Showing the startup-configuration in JSON format:

```
switch# show startup-config json
Startup configuration:
{
    "AAA_Server_Group": {
        "local": {
            "group_name": "local"
        },
        "none": {
            "group_name": "none"
        }
    },
...
```

show system error-counter-monitor

Syntax

`show system error-counter-monitor {basic <PORT-NUM> | extended} [vsx-peer]`

Description

Shows error counter statistics.

Command context

Manager (#)

Parameters

`basic <PORT-NUM>`

Specifies a physical port on the switch. Use the format `member/slot/port` (for example, `1/3/1`).

`extended`

Shows statistics for all interfaces.

Command context

Manager (`#`)

Examples

Showing error counter statistics for interface 1/1/1:

```
switch# show system error-counter-monitor basic 1/1/1

Interface error counter statistics for 1/1/1

Error Counter                        Value
---------------------------------------
EtherStatsOversizePkts               983
EtherStatsUndersizePkts              1024
EtherStatsJabbers                    10
Dot3StatsAlignmentErrors             462
Dot3StatsFCSErrors                   321
Dot3StatsLateCollisions              2024
EtherStatsFragments                  121
Dot3StatsExcessiveCollisions         1025
IfInBroadcastPkts                    2001
```

Showing error counter statistics for all interfaces:

```
switch# show system error-counter-monitor extended

Interface error counter statistics for 1/1/1

Error Counter                        Value
---------------------------------------
EtherStatsOversizePkts               983
EtherStatsUndersizePkts              1024
EtherStatsJabbers                    10
Dot3StatsAlignmentErrors             462
Dot3StatsFCSErrors                   321
Dot3StatsLateCollisions              2024
EtherStatsFragments                  121
Dot3StatsExcessiveCollisions         1025
IfInBroadcastPkts                    2001
...
...
Interface error counter statistics for 1/8/32

Error Counter                       Value
---------------------------------------
EtherStatsOversizePkts               0
...
```

show system

Syntax

```
show system [vsx-peer]
```

Description

Shows general status information about the system.

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.

Usage

CPU utilization represents the average utilization across all the CPU cores.

System Contact, System Location, and System Description can be set with the `snmp-server` command.

Examples

Showing system information on a 6300 switch:

```
switch(config)# show system
Hostname           : switch
System Description : FL.10.xx.xxxxx
System Contact     :
System Location    :

Vendor             : Aruba
Product Name       : JL659A 6300M 48SR5 CL6 PoE 4SFP56 Swch
Chassis Serial Nbr : ID9ZKHN090
Base MAC Address   : 9020c2-245080
ArubaOS-CX Version : FL.10.xx.xxxx

Time Zone          : UTC

Up Time            : 5 days, 15 hours, 33 minutes
CPU Util (%)       : 21
Memory Usage (%)   : 19
```

Showing system information om a 6400 switch:

```
switch(config)# show system
Hostname           : switch
System Description : FL.10.xx.xxxxx
System Contact     :
System Location    :

Vendor             : Aruba
Product Name       : R0X24A 6405 Chassis
```

```
Chassis Serial Nbr : SG9ZKM7206
Base MAC Address   : 9020c2-dc4700
ArubaOS-CX Version : FL.10.xx.xxxxx

Time Zone          : UTC

Up Time            : 32 minutes
CPU Util (%)       : 3
Memory Usage (%)   : 10
BLDG02-F3(config)#
```

show system resource-utilization

Syntax

```
show system resource-utilization [daemon <DAEMON-NAME> | all | module <SLOT-ID> | standby]
[vsx-peer]
```

Description

Shows information about the usage of system resources such as CPU, memory, and open file descriptors.

Command context

Manager (#)

Parameters

`daemon <DAEMON-NAME>`

Shows the filtered resource utilization data for the process specified by *<DAEMON-NAME>* only.

For a list of daemons that log events, enter `show events -d ?` from a switch prompt in the manager (#) context.

`all`

Shows utilization information for all VSF members.

`module <SLOT-ID>`

Shows the filtered resource utilization data for the line module specified by *<SLOT-ID>* only.

`[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

Showing all system resource utilization data:

```
switch# show system resource-utilization
System Resources:
Processes: 70
CPU usage(%): 20
Memory usage(%): 25
Open FD's: 1024
```

```
Process              CPU Usage(%)         Memory Usage(%)     Open FD's
-------------------------------------------------------------------------
pmd                     2                       1                 14
hpe-sysmond             1                       2                 11
hpe-mgmdd               0                       1                 5
...
```

Showing the resource utilization data for the pmd process:

```
switch# show system resource-utilization daemon pmd
Process              CPU Usage         Memory Usage    Open FD's
-------------------------------------------------------------------------
pmd                     2                1              14
```

Showing resource utilization data when system resource utilization polling is disabled:

```
switch# show system resource-utilization
System resource utilization data poll is currently disabled
```

Showing resource utilization data for a line module:

```
switch# show system resource-utilization module 1/1
System Resource utilization for line card module: 1/1
CPU usage(%): 0
Memory usage(%): 35
Open FD's: 512
```

show tech

Syntax

`show tech [basic | <FEATURE>] [local-file]`

Description

Shows detailed information about switch features by automatically running the `show` commands associated with the feature. If no parameters are specified, the `show tech` command shows information about all switch features. Technical support personnel use the output from this command for troubleshooting.

Command context

Manager (`#`)

Parameters

`basic`

Specifies showing a basic set of information.

<FEATURE>

Specifies the name of a feature. For a list of feature names, enter the `show tech` command, followed by a space, followed by a question mark (?).

`local-file`

Shows the output of the `show tech` command to a local text file.

Authority

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

Usage

To terminate the output of the `show tech` command, enter **Ctrl+C**.

If the command was not terminated with **Ctrl+C**, at the end of the output, the `show tech` command shows the following:

- The time consumed to execute the command.
- The list of failed `show` commands, if any.

To get a copy of the local text file content created with the show tech command that is used with the local-file parameter, use the `copy show-tech local-file` command.

Example

Showing the basic set of system information:

```
switch# show tech basic
================================================================
Show Tech executed on Wed Sep  6 16:50:37 2017
================================================================
================================================================
[Begin] Feature basic
================================================================

******************************
Command : show core-dump all
******************************
no core dumps are present

...
================================================================
[End] Feature basic
================================================================

================================================================
1 show tech command failed
================================================================
Failed command:
  1. show boot-history
================================================================
Show tech took 3.000000 seconds for execution
```

Directing the output of the **show tech basic** command to the local text file:

```
switch# show tech basic local-file
Show Tech output stored in local-file. Please use 'copy show-tech local-file'
to copy-out this file.
```

show usb

Syntax

```
show usb [vsx-peer]
```

Description

Shows the USB port configuration and mount settings.

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

If USB has not been enabled:

```
switch> show usb
Enabled: No
Mounted: No
```

If USB has been enabled, but no device has been mounted:

```
switch> show usb
Enabled: Yes
Mounted: No
```

If USB has been enabled and a device mounted:

```
switch> show usb
Enabled: Yes
Mounted: Yes
```

show usb file-system

Syntax

`show usb file-system [<PATH>]`

Description

Shows directory listings for a mounted USB device. When entered without the <PATH> parameter the top level directory tree is shown.

Command context

Operator (>) or Manager (#)

Parameters

<PATH>

Specifies the file path to show. A leading "/" in the path is optional.

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

Adding a leading "/" as the first character of the *<PATH>* parameter is optional.

Attempting to enter '..' as any part of the *<PATH>* will generate an invalid path argument error. Only fully-qualified path names are supported.

Examples

Showing the top level directory tree:

```
switch# show usb file-system
/mnt/usb:
'System Volume Information'  dir1'

/mnt/usb/System Volume Information':
IndexerVolumeGuid  WPSettings.dat

/mnt/usb/dir1:
dir2  test1

/mnt/usb/dir1/dir2:
test2
```

Showing available path options from the top level:

```
switch# show usb file-system /
total 64
drwxrwxrwx 2 32768 Jan 22 16:27 'System Volume Information'
drwxrwxrwx 3 32768 Mar  5 15:26 dir1
```

Showing the contents of a specific folder:

```
switch# show usb file-system /dir1
total 32
drwxrwxrwx 2 32768 Mar  5 15:26 dir2
-rwxrwxrwx 1     0 Feb  5 18:08 test1

switch# show usb file-system dir1/dir2
total 0
-rwxrwxrwx 1 0 Feb  6 05:35 test2
```

Attempting to enter an invalid character in the path:

```
switch# show usb file-system dir1/../..
Invalid path argument
```

show version

Syntax

```
show version [vsx-peer]
```

Description

Shows version information about the network operating system software, service operating system software, and BIOS.

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

Showing version information for a 6300 switch:

```
switch(config)# show version
--------------------------------------------------------------------------------
ArubaOS-CX
(c) Copyright 2017-2020 Hewlett Packard Enterprise Development LP
--------------------------------------------------------------------------------
Version      : FL.xx.xx.xxxx
Build Date   : 2020-10-22 17:00:46 PDT
Build ID     : ArubaOS-CX:FL.xx.xx.xxxx:85c3c2f3d59e:201910222335
Build SHA    : 85c3c2f3d59ec8318ba97178fad387aecb671b33
Active Image : secondary

Service OS Version : FL.01.05.0001-internal
BIOS Version       : FL.01.0001
```

system resource-utilization poll-interval

Syntax

```
system resource-utilization poll-interval <SECONDS>
```

Description

Configures the polling interval for system resource information collection and recording such as CPU and memory usage.

Command context

```
config
```

Parameters

<SECONDS>

Specifies the poll interval in seconds. Range: 10-3600. Default: 10.

Authority

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

Example

Configuring the system resource utilization poll interval:

```
switch(config)# system resource-utilization poll-interval 20
```

top cpu

Syntax

```
top cpu
```

Description

Shows CPU utilization information.

Command context

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 top CPU information:

```
switch# top cpu
top - 09:42:55 up 3 min, 3 users, load average: 3.44, 3.78, 1.70
Tasks:  76 total, 2 running, 74 sleeping,  0 stopped,  0 zombie
%Cpu(s): 31.4 us, 32.7 sy, 0.5 ni, 34.4 id, 04. wa, 0.0 hi, 0.6 si, 0.0 st
KiB Mem : 4046496 total,  2487508 free,  897040 used,   661948 buff/cache
KiB Swap:       0 total,        0 free,       0 used,  2859196 avail Mem

  PID USER     PRI  NI   VIRT    RES   SHR S  %CPU %MEM      TIME+ COMMAND
...
```

top memory

Syntax

```
top memory
```

Description

Shows memory utilization information.

Command context

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 top memory:

```
switch> top memory
top - 09:42:55 up 3 min, 3 users, load average: 3.44, 3.78, 1.70
Tasks:  76 total, 2 running, 74 sleeping,  0 stopped,  0 zombie
%Cpu(s): 31.4 us, 32.7 sy, 0.5 ni, 34.4 id, 04. wa, 0.0 hi, 0.6 si, 0.0 st
KiB Mem : 4046496 total,  2487508 free,  897040 used,   661948 buff/cache
KiB Swap:       0 total,        0 free,       0 used,  2859196 avail Mem

  PID USER     PRI  NI    VIRT    RES    SHR S  %CPU %MEM      TIME+ COMMAND
...
```

usb

Syntax

```
usb
no usb
```

Description

Enables the USB ports on the switch. This setting is persistent across switch reboots and management module failovers. Both active and standby management modules are affected by this setting.

The `no` form of this command disables the USB ports.

Command context

```
config
```

Authority

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

Example

Enabling USB ports:

```
switch(config)# usb
```

Disabling USB ports when a USB drive is mounted:

```
switch(config)# no usb
```

usb mount | unmount

Syntax

```
usb {mount | unmount}
```

Description

Enables or disables the inserted USB drive.

Command context

Manager (#)

Parameters

`mount`

Enables the inserted USB drive.

`unmount`

Disables the inserted USB drive in preparation for removal.

Authority

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

Usage

If USB has been enabled in the configuration, the USB port on the active management module is available for mounting a USB drive. The USB port on the standby management module is not available.

An inserted USB drive must be mounted each time the switch boots or fails over to a different management module.

A USB drive must be unmounted before removal.

The supported USB file systems are FAT16 and FAT32.

Examples

Mounting a USB drive in the USB port:

```
switch# usb mount
```

Unmounting a USB drive:

```
switch# usb unmount
```

Accessing Aruba Support

Aruba Support Services	https://www.arubanetworks.com/support-services/
Aruba Support Portal	https://asp.arubanetworks.com/
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	https://www.arubanetworks.com/support-services/contact-support/

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	https://community.arubanetworks.com/
Software licensing	https://lms.arubanetworks.com/
End-of-Life information	https://www.arubanetworks.com/support-services/end-of-life/
Aruba software and documentation	https://asp.arubanetworks.com/downloads

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). 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.