AOS-CX 10.09 Job Scheduler Guide

4100i, 6000, 6100, 6200, 6300, 6400, 8320, 8325, 8360, 8400 Switch Series

a Hewlett Packard Enterprise company

Published: September 2022
Edition: 3

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.

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.

- The Job Scheduler enables you to execute batches of CLI commands on a user-configured schedule or interval. Job Scheduler can be used, for example, to schedule activities such as port toggles, switch reboots, QoS policy changes, system health status checks, statistics clearing, clean-up, and saving the running configuration.
- Schedules can trigger jobs based on calendar date and time or at periodic intervals.
- Jobs can be scheduled to execute as frequently as once every thirty minutes.
- When executed, commands with simple (y/n) prompts (such as `boot system`) will be automatically confirmed with "y." Other commands requiring more complex user input (such as password change) cannot be used.

Working with Job Scheduler

To help understand how to work with the Job Scheduler, several basic examples are presented, followed by detailed descriptions of the commands involved under Job scheduler commands.

Port toggle example

This example creates a port toggle job and then schedules the job for execution on Monday and Friday night at 11:45 PM.

Creating a port toggle job named **PTog1**:

```
switch(config)# job PTog1
switch(config-job-PTog1)# desc Toggle port 1/1/1
switch(config-job-PTog1)# 10 cli config
switch(config-job-PTog1)# 20 cli interface 1/1/1
switch(config-job-PTog1)# 30 cli shutdown
switch(config-job-PTog1)# 40 delay 10 cli no shutdown
switch(config-job-PTog1)# 50 cli end
switch(config-job-PTog1)# exit
```

Creating a schedule named **PT2xW** that runs the port toggle job **PTog1** on Mondays and Fridays at 11:45 PM, starting on August 2 2021, with a one-year duration:

```
switch(config)# schedule PT2xW
switch(config-schedule-PT2xW)# desc Monday & Friday 11:45 PM port toggles
switch(config-schedule-PT2xW)# 10 job PTog1
switch(config-schedule-PT2xW)# trigger on 23:45 weekly 2,6 count 104 start 2021-08-02
switch(config-schedule-PT2xW)# exit
```

Showing the port toggle job information after first execution:

```
switch# show job PTog1
```

```
Job Name : PTog1

    Enabled                : Yes
    Description            : Toggle port 1/1/1
    Status                 : waiting
    Number of commands     : 5
    Total execution count  : 1
    Failed execution count : 0

    Job execution history
    ---------------------

    Instance number        : 1
    Execution status       : success
    Execution start time   : Mon Aug 2 23:45:00 2021
    Execution duration     : 10s

    Job CLI commands
    ----------------
    10 cli config
    20 cli interface 1/1/1
    30 cli shutdown
    40 delay 10 cli no shutdown
    50 cli end
```

Showing the port toggle job schedule information after first execution:

```
switch# show schedule PT2xW

Schedule Name: PT2xW

    Schedule config
    ---------------
    Description         : Monday & Friday 11:45 PM port toggles
    Enabled             : Yes
    Trigger type        : calendar
    Transient           : No
    Max trigger count   : 104
    Trigger start date  : 2021-08-02 23:45

    Schedule Status
    ---------------
    Trigger status      : active
    Next trigger time   : Fri Aug  6 23:45:00 2021
    Triggered count     : 1

    Scheduled Jobs
    --------------
    10  : PTog1
```

Showing the port toggle job most recent execution output:

```
switch# show job PTog1 execution-output 1
================================================================================================
Command: config
time: Mon Aug  2 23:45:00 2021
================================================================================================

================================================================================================
```

```
Command: interface 1/1/1
time: Mon Aug  2 23:45:00 2021
================================================================================

================================================================================
Command: shutdown
time: Mon Aug  2 23:45:00 2021
================================================================================

================================================================================
Command: cli no shutdown
time: Mon Aug  2 23:45:10 2021
================================================================================

================================================================================
Command: end
time: Mon Aug  2 23:45:10 2021
================================================================================
```

Switch reboot example

This example creates a switch reboot job and then schedules the job for execution on the last day of every month at 3:00 AM.

Creating a job named **Reboot_sw1** that saves the running configuration and then reboots the switch:

```
switch(config)# job Reboot_sw1
switch(config-job-Reboot_sw1)# desc Save config then reboot switch
switch(config-job-Reboot_sw1)# 10 cli config
switch(config-job-Reboot_sw1)# 20 cli write mem
switch(config-job-Reboot_sw1)# 30 cli boot system
switch(config-job-Reboot_sw1)# exit
switch(config)#
```

Creating a schedule named **RB_LDM** that runs the switch reboot job **Reboot_sw1** on the last day of the month at 3:00 AM, starting on January 31 2022, with a two-year duration:

```
switch(config)# schedule RB_LDM
switch(config-schedule-RB_LDM)# desc Monthly reboot 3:00 AM
switch(config-schedule-RB_LDM)# 10 job Reboot_sw1
switch(config-schedule-RB_LDM)# trigger on 3:00 monthly 31 count 24 start 2022-01-31
switch(config-schedule-RB_LDM)# exit
switch(config)#
```

After the **RB_LDM** schedule triggers the reboot job **Reboot_sw1**, the `show events` command is available to show schedule triggering (*<MODEL>* represents the switch model number):

```
switch# show events -a -d schedulerd
-------------------------------------------------
Event logs from previous boots
-------------------------------------------------
...
2022-01-31T03:00:14.405135+00:00 <MODEL> schedulerd[2054]: Event|12202|LOG_
INFO|AMM|1/1|Schedule RB_LDM triggered, trigger_count: 1
-------------------------------------------------
Event logs from current boot
```

```
-------------------------------------------------
switch#
```