Hewlett Packard Enterprise

HPE Cray XD670 Ansible Firmware Update Tool User Guide

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

Intel®, Itanium®, Optane™, Pentium®, Xeon®, Intel Inside®, and the Intel Inside logo are trademarks of Intel Corporation or its subsidiaries.

AMD and the AMD EPYC™ and combinations thereof are trademarks of Advanced Micro Devices, Inc.

Microsoft® and Windows® are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Adobe® and Acrobat® are trademarks of Adobe Systems Incorporated.

Java® and Oracle® are registered trademarks of Oracle and/or its affiliates.

UNIX® is a registered trademark of The Open Group.

All third-party marks are property of their respective owners.

Table of Contents

Overview

The HPE Cray XD Ansible Firmware Update Tool (AFUT) provides a mechanism to quickly update the firmware components of HPE Cray XD Server nodes, whether individually or many at a time. It has support for the XD670 models. This Ansible based tool can be executed from Linux either from management or administrator nodes to update the components. This tool can also be used to create firmware inventory reports for HPC cluster nodes. It is necessary to install a full Ansible environment and other prerequisites on the management workstation as it is an Ansible script-based tool.

Supported operating system

The tool supports execution from Linux operating system.

1. Ubuntu (22.04)

2. Red Hat Enterprise Linux (RHEL) (RHEL 8, and RHEL 9)

3. SLES 15 SP5 (Supporting all firmware updates except GPU)

Prerequisites

1. Ensure that Ansible is installed.

2. The Ansible collection community.general version 8.2.0 or later must be installed.

3. To install the package, use:

    i. sudo apt install ansible

    ii. ansible-galaxy collection install community.general

To install all the remaining prerequisites, use the setup.yml file based on operating system

4. Ensure request toolbelt version 0.9.1 or later is installed.

    i. To install, run pip install requests-toolbelt

5. GPU update requirement:

    ◦ **BMC Firmware**: Minimum version 1.09 or later is required.

    ◦ **BPB CPLD Firmware**: Minimum version 3:04 or later is required.

    ◦ Minimum 120 MB of free memory is required in BMC for the GPU update (Reset BMC through Redfish and Power Cycle if required memory is not available)

        • To check free memory, use the below Redfish URI

            "/redfish/v1/UpdateService/Action/Oem/Gbt/HMCUpdate.PrepareFreeMemory"

Supported Models

1. HPE Cray XD670 – Gigabyte using Aspeed/AMI BMC firmware

Supported Targets for Update

- HPE Cray XD670

    - BMC

    - BMCImage2 (This component can be updated only in Non-RoT servers)

    - BIOS

    - BIOS2 (This component can be updated only in Non-RoT servers)

    - BPB_CPLD

    - MB_CPLD1_SCM_CPLD1

    - GPU_ALL

Install AFUT

1. You can download AFUT by cloning this repository: https://github.com/HewlettPackard/CrayXD_AFUT_XD670

2. Install and unzip the Ansible_Firmware_Update_Tool.zip package at the Management Workstation where you want to deploy the firmware.

3. The unzipped folder contains:

    - `system_firmware_update.yml` – The main Ansible playbook file that contains tasks responsible for flashing firmware.

    - `get_system_firmware_inventory.yml` – The Ansible playbook file, which is responsible for report generation with firmware inventory information.

    - `get_gpu_inventory.yml` – The Ansible playbook file, which is responsible for report generation of the GPU inventory information.

    - `power_state_XD670.yml` – The Ansible playbook is used to fetch and change the power states of the HPE Cray XD670.

    - `Inventory` – Create an Ansible inventory file that lists the target systems you want to update or retrieve inventory details. This file should contain the IP addresses to connect to the systems.

    - `system_credentials.yml` – This file contains the credentials of the target systems.

    - `config.ini` – Configuration file contains four sections Target, Image, Power and Firmware Type. Under these sections, options must be filled with valid keys.

    - `ubuntu_setup.yml` - This playbook is used to install prerequisites on the Ubuntu system. Use the following command to run this playbook:

        ```
        $ ansible-playbook ubuntu_setup.yml --ask-become-pass -i inventory
        ```

    - `RHEL_setup.yml` – This playbook is used to install prerequisites on the RHEL system. Use the following command to run this playbook:

        ```
        $ ansible-playbook RHEL_setup.yml --ask-become-pass -i inventory
        ```

- `SUSE_setup.yml` – This playbook is used to install prerequisites on the SUSE system. Use the following command to run this playbook:

  ```
  $ ansible-playbook SUSE_setup.yml --ask-become-pass -i inventory
  ```

Notes

- Enter the correct details in config.ini file to start update. If wrong information is given, AFUT might not be able to detect the firmware file of the update or might print the error message on the command line.

- For HPE Cray XD670, the SCM CPLD and MB CPLD must be upgraded together. Do not perform an AC power cycle without updating both CPLD firmware files. If a power cycle is done in between the update, it might prevent the system from powering back on. The target MB_CPLD1_SCM_CPLD1 is combination of MB CPLD and SCM CPLD and must be given during the update.

- The Power State must be on for BPB_CPLD and SCM_CPLD before the update. The power_state_XD670.yml playbook can be used to switch the power state to on or else the update will be skipped.

- After the update of BPB_CPLD, SCM and MB_CPLD is completed in pair, it is mandatory to plug out and plug in the power manually.

- IPV6 addresses are not supported with the current AFUT. IP addresses refer to IPv4 only and not IPv6.

- Check the generated CSV output file to confirm the firmware update status.

- AFUT does not have any limitation on the number of nodes. As it is not mentioned in any Ansible documentation, see https://docs.ansible.com/.

- For GPU updates, AFUT will only trigger the update. User has to monitor the update progress using Redfish. Once complete, the updated version can be verified using the "get_gpu_inventory.yml" playbook.

- If BIOS firmware upgrade and downgrade fails from AFUT (between 1.xx and 2.xx), then try updating the BIOS firmware through GUI and ensure that the "Preserve configuration settings" option is not enabled. (This enhancement will be included in the later release of AFUT so that it's automatically taken care for multiple systems.)

Firmware files

Download the latest component firmware packs from the HPE Support Centre.

Supported files for the update:

- Use HPM applications or octet-stream files to flash BMC and BIOS:

- For GPU updates, use .fwpkg files.

- CPLD firmware updates require .RCU files for the update

Ensure that the file is present on the local machine and provide the correct file path in the configuration file for execution.

Configuration

- `system_credentials.yml` contains the credentials of the server in the form of its username and password.

An example of `system_credentials.yml` is as follows:
```
---
inputs:

 10.xx.xx.xx:
```

```
    user: "username"
    password: "password"
 10.xx.xx.xx:
    user: "username"
    password: "password"
 10.xx.xx.xx:
    user: "username"
    password: "password"
```

- `inventory.txt` – inventory.txt file contains the details of the servers IP, mention all the IP that must be used for execution. If an IP is marked with "#", then it will not be considered for execution.

An example of `inventory` is as follows:

```
[xds]
#IP1
IP2
IP3
```

- `config.ini` – This configuration file maintains the Target, Image, Options (for power state) and Firmware type for the firmware update.

Following is an example of the configuration file:

```
$ cat config.ini
```

```
[Image]
```

```
update_image_path_xd670 =
```

```
[Target]
```

```
update_target =
```

```
##Allowed options are:
```

```
#XD670 - BMC, BMCImage2, BIOS, BIOS2, SCM_CPLD1_MB_CPLD1, BPB_CPLD, GPU_ALL
```

```
##BMC is equivalent to BMCImage1 here for Cray XD670 systems with Non-RoT motherboards (BMC
remains same as there is only a single target which is equal to BMC for RoT motherboards)
```

```
##BPB_CPLD is the update of BPB_CPLD1 and BPB_CPLD2 firmwares back to back, it requires only one
hpm file
```

```
##SCM_CPLD1_MB_CPLD1 is the update of SCM_CPLD1 and MB_CPLD1 firmwares back to back, update files
for the same should be mentioned as
```

```
#SCM_CPLD1_file.hpm MB_CPLD1_file.hpm
```

```
##first file to be of SC_CPLD1 and space and followed by MB_CPL1 hpm file
```

```
##GPU_ALL is to update all components in the GPU (Firmware type should be HMC)
```

```
##all targets are case-sensitive. follow the exact same cases.
```

```
[Options]
```

```
power_state =
```

```
##Allowed options are: NA on off
```

```
[Firmware_type]
update_image_type =
##Allowed options(Default: HPM):
#Mention firmware type only for below components
#HPM : For BMC, BIOS and CPLD components(MB_CPLD, SCM_CPLD, BPB_CPLD)
#HMC : For complete GPU update
```

Ansible Vault Implementation

Ansible-Vault helps to encrypt files using a password, here we encrypt our system credentials file using a password to make it more secure.

Following are examples for using an Ansible vault

- `cat system_credentials.yml`
```
---
inputs:
 10.xx.xx.xx:
    user: "username"
    password: "password"
 10.xx.xx.xx:
    user: "username"
    password: "password"
 10.xx.xx.xx:
    user: "username"
    password: "password"
```

- `ansible-vault encrypt system_credentials.yml`
```
New Vault password:
Confirm New Vault password:
Encryption successful
```

- `cat system_credentials.yml`
```
$ANSIBLE_VAULT;1.1;AES256
39313431623835643333062393539623262376236623738646231363165653234303134353966564
33363963613634623033633565383139353539633032653406373563656536343136393731616531
66313939633035663734346537303332393033396364316537613626238376365396438343761363
38646165373165636510a6263332656465537323630373065363436303337623765666162333339
37623264376637633262303839386369663439383161353435333353265306136373731336230331
66633962316238336239373535462323063316463338332939373162316530383934663035643735
39323662633432239393261373039913565343063656365313931626530613833346635306563631
36623231653738363334333388313635326164333961313936633035643364333338656533263346338
64313235343437653663376535562666165366637326536262163338376139663231643531656337309
65353937323736353031313237353863366332623732636334366631303961613431303430626437

38643433666163333306534306331623039383232363562623861376536373639396163643430381
61663939623430613336346562653313361623237323237373438031643734323264353133396635
6264316139363862343066356630393932653633346338437834363832383665633964323435
62393864633964323366653461646662396238376437383637366
```

- `ansible-vault edit system_credentials.yml`
```
Vault password:
```

- `ansible-vault view system_credentials.yml`
```
Vault password:
- --
inputs:

 10.xx.xx.xx:
```

```
     user: "username"
     password: "password"
 10.xx.xx.xx:
     user: "username"
     password: "password"
 10.xx.xx.xx:
     user: "username"
     password: "password"
```

- `cat system_credentials.yml`

```
$ANSIBLE_VAULT;1.1;AES256
39313431623835643330623935396232623762366237386462313631656532343031343539666564
33363936361363462303363356538313935353963303265340a37356365653634313639373161653 1
66313939633035663734346537303332393033396364316537613262238376365396438343761336 3
38646165373165636 10a623633326564653537323630373065363436303337623765666162333339
376232643766376332623038393 383639663439383 16135343533335326530613637373 13362303331
66339623162383362393735346232306331643633383332393731623165303839346630356437 35
39323662633432393932613730393 1356534306365636531393 162653061383334663530656366 31
63623231653738363334333383136353261643339613 1393663303564336433338656532633463 38
6431323534343765366337653356266616536663732653662616338376 1396632316435316563370
6535393732373635303131323735386336633262373263363343666313039616134313303430626437
38643436661363330653430633162303938323236526238613765363736393961636434303831
61663939623430613336364562653133616232373232373734383031643734323264353133396635
62643 16139363862343066356630393932653633346338343738343638323836563396430323435
6239386463396432336665346164666239623837643738363736
```

- `ansible-vault decrypt system_credentials.yml`
```
Vault password:
Decryption successful
```

- `cat system_credentials.yml`
```
---
inputs:
 10.xx.xx.xx:
     user: "username"
     password: "password"
 10.xx.xx.xx:
     user: "username"
     password: "password"
 10.xx.xx.xx:
     user: "username"
     password: "password"
```

Below is an example for usage of encrypted file in command execution. "–ask-vault-pass" needs to be used while execution of command if any of the file is encrypted. Vault password is needed during the execution:

```
$ ansible-playbook get_system_firmware_inventory.yml -i inventory --ask-vault-pass -e
@system_credentials.yml
```

```
Vault password:
```

Below is a similar example for "`system_firmware_update.yml`", while it's execution with encrypted "system_credentials.yml ".

```
$ ansible-playbook system_firmware_update.yml -i inventory --ask-vault-pass -e
@system_credentials.yml
```

```
Vault password:
```

Commands

usage: ansible-playbook: [-e EXTRA_VARS] [-i INVENTORY] [--ask-vault-password]

options:
  --ask-vault-password, --ask-vault-pass
                ask for vault password
-e EXTRA_VARS, --extra-vars EXTRA_VARS
                set additional variables as key=value or YAML/JSON, if filename prepend with @
  -i inventory, --inventory inventory, --inventory-file inventory
                specify inventory host path or comma separated host list. --inventory-file is deprecated

Running the Utility for Report

The "get_system_firmware_inventory.yml" playbook generates a report containing the IP details, models, and component versions, saved in a CSV file.
Before execution, ensure that the system credentials and inventory details are correctly filled in their respective files

Following command is used for the execution of Inventory file:

```
$ ansible-playbook -i inventory get_system_firmware_inventory.yml -e @system_credentials.yml

PLAY [version 1.0 Fetches the AFUT supported Cray XD servers System Firmware Inventory Details
along with Model name] ******************************************************************

TASK [Gathering Facts]
****************************************************************************************
*****************************************************************
ok: [10.xx.xx.xx]

TASK [All System Firmware Inventory Details will be stored in the below csv file]
****************************************************************************************
*********
ok: [10.xx.xx.xx]

TASK [Fetching System Firmware Inventory Details]
****************************************************************************************
*****************************************
[WARNING]: Collection community.general does not support Ansible version 2.10.8
ok: [10.xx.xx.xx]


TASK [Writing inventory details to All_System_FW_Inventory_2024-07-31_21:42:19.csv file]

****************************************************************************************
**
changed: [10.xx.xx.xx]

PLAY RECAP

****************************************************************************************
************************************************************************
10.xx.xx.xx              : ok=4    changed=1    unreachable=0    failed=0    skipped=0

rescued=0    ignored=0
```

Following is an example output of a CSV file:

```
$ cat All_System_FW_Inventory_2024-07-31_21:42:19.csv
IP_Address,Model,BMC,BMCImage1,BMCImage2,BIOS,BIOS2,BPB_CPLD1,BPB_CPLD2,MB_CPLD1,SCM_CPLD1
10.xx.xx.xx, HPE Cray SC XD670 DLC,1.17.00,NA,NA,CUXD670_5.32_v2.00,NA ,05,32,22,10
```

The "get_gpu_inventory.yml" playbook generates a report consisting of the IP details, Models and the Components version of GPU in a csv file.

Following command is used for the execution of GPU Inventory file:

```
$ ansible-playbook -i inventory get_gpu_inventory.yml -e @system_credentials.yml

PLAY [version 1.0 Fetches the AFUT supported Cray XD servers GPU Inventory Details along with
Model name] ********************************************************************

TASK [Gathering Facts]
*****************************************************************************************
*********************************************************************
ok: [10.xx.xx.xx 1]

TASK [All GPU Inventory Details will be stored in the below csv file]
*****************************************************************************************
*********
ok: [10.xx.xx.xx]

TASK [Fetching GPU Inventory Details]
*****************************************************************************************
******************************************
[WARNING]: Collection community.general does not support Ansible version 2.10.8
ok: [10.xx.xx.xx]

TASK [Writing GPU inventory details to GPU_FW_Inventory_2024-06-26_03:05:27.csv file]
*****************************************************************************************
*********
changed: [10.xx.xx.xx]

PLAY RECAP
*****************************************************************************************
*******************************************************************
10.xx.xx.xx                 : ok=4    changed=1    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0

$ cat  GPU_FW_Inventory_2024-06-26_03:05:27.csv

IP_Address,Model,HGX_FW_BMC_0,HGX_FW_ERoT_BMC_0,HGX_FW_ERoT_FPGA_0,HGX_FW_ERoT_GPU_SXM_1,HGX_FW_ER
oT_GPU_SXM_2,HGX_FW_ERoT_GPU_SXM_3,HGX_FW_ERoT_GPU_SXM_4,HGX_FW_ERoT_GPU_SXM_5,HGX_FW_ERoT_GPU_SXM
_6,HGX_FW_ERoT_GPU_SXM_7,HGX_FW_ERoT_GPU_SXM_8,HGX_FW_ERoT_NVSwitch_0,HGX_FW_ERoT_NVSwitch_1,HGX_F
W_ERoT_NVSwitch_2,HGX_FW_ERoT_NVSwitch_3,HGX_FW_ERoT_PCIeSwitch_0,HGX_FW_FPGA_0,HGX_FW_GPU_SXM_1,H
GX_FW_GPU_SXM_2,HGX_FW_GPU_SXM_3,HGX_FW_GPU_SXM_4,HGX_FW_GPU_SXM_5,HGX_FW_GPU_SXM_6,HGX_FW_GPU_SXM
_7,HGX_FW_GPU_SXM_8,HGX_FW_NVSwitch_0,HGX_FW_NVSwitch_1,HGX_FW_NVSwitch_2,HGX_FW_NVSwitch_3,HGX_FW
_PCIeRetimer_0,HGX_FW_PCIeRetimer_1,HGX_FW_PCIeRetimer_2,HGX_FW_PCIeRetimer_3,HGX_FW_PCIeRetimer_4
,HGX_FW_PCIeRetimer_5,HGX_FW_PCIeRetimer_6,HGX_FW_PCIeRetimer_7,HGX_FW_PCIeSwitch_0,HGX_InfoROM_GP
U_SXM_1,HGX_InfoROM_GPU_SXM_2,HGX_InfoROM_GPU_SXM_3,HGX_InfoROM_GPU_SXM_4,HGX_InfoROM_GPU_SXM_5,HG
X_InfoROM_GPU_SXM_6,HGX_InfoROM_GPU_SXM_7,HGX_InfoROM_GPU_SXM_8
10.xx.xx.xx,HPE Cray SC XD670 DLC,HGX-22.10-1-
rc63,00.02.0178.0000_n00,00.02.0178.0000_n00,00.02.0178.0000_n00,NA,00.02.0178
.0000_n00,00.02.0178.0000_n00,00.02.0178.0000_n00,00.02.0178.0000_n00,00.02.0178.0000_n00,00.02.01
78.0000_n00,00.02.0178.0000_n00,00.02.0178.0000_n00,00.02.0178.0000_n00,00.02.0178.0000_n00,2.41,9
6.00.99.00.0D,96.00.99.00.0D,96.00.99.00.0D,96.00.99.00.0D,96.00.99.00.0D,96.00.99.00.0D,96.00.99.
00.0D,96.00.99.00.0D,96.10.52.00.01,96.10.52.00.01,96.10.52.00.01,96.10.52.00.01,2.7.5,2.7.5,2.7.5
```

```
,2.7.5,2.7.5,2.7.5,2.7.5,2.7.5,1.9.5F,G520.0202.00.03,G520.0202.00.03,G520.0202.00.03,G520.0202.00
.03,G520.0202.00.03,G520.0202.00.03,G520.0202.00.03,G520.0202.00.03
```

Running the Utility for Power State

Following command is used for the execution of Power State file:

To check the current state, the power state in the configuration file should be as 'NA', then to change the power state use 'on' or 'off' in the power state section of configuration file

```
$ ansible-playbook -i inventory power_state_XD670.yml -e @system_credentials.yml

PLAY [version 1.0 Get Power State of HPE Cray XD670 model nodes]
*************************************************************************************************
*************************

TASK [Gathering Facts]
*************************************************************************************************
*********************************************************************

ok: [10.xx.xx.xx]

TASK [Power states of the nodes will be uploaded to the below csv file]
*************************************************************************************************
*******************

ok: [10.xx.xx.xx]

TASK [Getting Power State of Cray XD670 Server nodes]
*************************************************************************************************
************************************

[WARNING]: Collection community.general does not support Ansible version 2.10.8
ok: [10.xx.xx.xx]

TASK [Writing Power status details to Power_State_CrayXD670_2024-08-26_00:42:05.csv file]
*************************************************************************************************
*

changed: [10.xx.xx.xx]

PLAY RECAP
*************************************************************************************************
********************************************************************************

10.xx.xx.xx                : ok=4    changed=1    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0

$ cat Power_State_CrayXD670_2024-08-26_00:42:05.csv
IP_Address,Model,Power_State
10.xx.xx.xx,HPE Cray XD670,On

$ cat config.ini
[Image]
update_image_path_xd670 =

[Target]
update_target =
```

```
##Allowed options are:
#XD670 - BMC, BMCImage2, BIOS, BIOS2, SCM_CPLD1_MB_CPLD1, BPB_CPLD, GPU_ALL
##BMC is equivalent to BMCImage1 here for Cray XD670 systems with Non-RoT motherboards (BMC
remains same as there is only a single target which is equal to BMC for RoT motherboards)
##BPB_CPLD is the update of BPB_CPLD1 and BPB_CPLD2 firmwares back to back, it requires only one
hpm file
##SCM_CPLD1_MB_CPLD1 is the update of SCM_CPLD1 and MB_CPLD1 firmwares back to back, update files
for the same should be mentioned as
#SCM_CPLD1_file.hpm MB_CPLD1_file.hpm ##first file to be of SC_CPLD1 and space and followed by
MB_CPL1 hpm file
##GPU_ALL is to update all components in the GPU (Firmware type should be HMC)
##all targets are case-sensitive. Please follow the exact same cases.

[Options]
power_state = off
##Allowed options are: NA on off
##applicable only for XD670

[Firmware_type]
update_image_type =
##Allowed options(Default: HPM):
#Mention firmware type only for below components
#HPM : For BMC, BIOS and CPLD components(MB_CPLD, SCM_CPLD, BPB_CPLD)
#HMC : For complete GPU update

$ ansible-playbook -i inventory power_state_XD670.yml -e @system_credentials.yml

PLAY [version 1.0 Get Power State of HPE Cray XD670 model nodes]
*********************************************************************************************
************************

TASK [Gathering Facts]
*********************************************************************************************
********************************************************************
ok: [10.xx.xx.xx]

TASK [Power states of the nodes will be uploaded to the below csv file]
*********************************************************************************************
*******************
ok: [10.xx.xx.xx]

TASK [Getting Power State of Cray XD670 Server nodes]
*********************************************************************************************
**************************************
[WARNING]: Collection community.general does not support Ansible version 2.10.8

ok: [10.xx.xx.xx]

TASK [Writing Power status details to Power_State_CrayXD670_2024-08-26_00:42:44.csv file]
*********************************************************************************************
changed: [10.xx.xx.xx]

PLAY RECAP
*********************************************************************************************
```

```
******************************************************************************
10.xx.xx.xx               : ok=4    changed=1    unreachable=0    failed=0    skipped=0

rescued=0    ignored=0

$ cat Power_State_CrayXD670_2024-08-26_00:42:44.csv
IP_Address,Model,Power_State
10.xx.xx.xx,HPE Cray XD670,Off
```

Running the Utility for Firmware Update

The "`system_firmware_update.yml`" playbook is used to update HPC nodes. The available update targets depend on the server model, and the supported targets for each model can be found in the [supported targets section](#).

Before performing the update:
- Specify the image path, target, firmware file type details in the playbook.
- Update the inventory and system credentials input files with the respective details of target system.

After the update:
- The status of the update tasks will be saved in a CSV file in the current directory.

Here is an example of the update for HPE Cray XD670 BMCimage2 from version 1.14 to 1.17

```
$ cat config.ini

[Image]
update_image_path_xd670 = XD670_BMC_v1.17_backup_signed.bin.hpm
[Target]
update_target = BMCImage2
##Allowed options are:
#XD670 - BMC, BMCImage2, BIOS, BIOS2, SCM_CPLD1_MB_CPLD1, BPB_CPLD, GPU_ALL
##BMC is equivalent to BMCImage1 here for Cray XD670 systems with Non-RoT motherboards (BMC
remains same as there is only a single target which is equal to BMC for RoT motherboards)
##BPB_CPLD is the update of BPB_CPLD1 and BPB_CPLD2 firmwares back to back, it requires only one
hpm file
##SCM_CPLD1_MB_CPLD1 is the update of SCM_CPLD1 and MB_CPLD1 firmwares back to back, update files
for the same should be mentioned as
#SCM_CPLD1_file.hpm MB_CPLD1_file.hpm ##first file to be of SCM_CPLD1 and space and followed by
MB_CPL1 hpm file
##GPU_ALL is to update all components in the GPU (Firmware type should be HMC)

##all targets are case-sensitive. follow the exact same cases.

 [Options]
power_state = NA
#NA on off
##applicable only for Cray XD670

[Firmware_type]
update_image_type =
##Allowed options(Default: HPM):
#Mention firmware type only for below components
#HPM : For BMC, BIOS and CPLD components(MB_CPLD, SCM_CPLD, BPB_CPLD)
#HMC : For complete GPU update

$ ansible-playbook -i inventory system_firmware_update.yml -e @system_credentials.yml

PLAY [version 1.0 System Firmware Update for HPE Cray XD670 model systems]
```

```
*********************************************

TASK [Gathering Facts]
*********************************************************************************************************

********************************************************************************
ok: [10.xx.xx.xx

TASK [System Firmware Update Status result will be uploaded to the below csv file]
*********************************************************************************************************
********
ok: [10.xx.xx.xx]

TASK [Running Firmware Update for Cray XD Servers]
*********************************************************************************************************
*******************************************
[WARNING]: Collection community.general does not support Ansible version 2.10.8
ok: [10.xx.xx.xx]

TASK [Writing Firmware Upgrade status details to System_FW_Update_2024-07-31_15:03:40.csv file]
*********************************************************************************************************
changed: [10.xx.xx.xx]

PLAY RECAP
*********************************************************************************************************
**********************************************************************************

10.xx.xx.xx                 : ok=4    changed=1    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0
```

Following is an example output of a CSV file:

```
$ cat System_FW_Update_2024-07-31_15:03:40.csv

IP_Address,Model,BMCImage2_Pre_Ver,BMCImage2_Post_Ver,Update_Status
10.xx.xx.xx,HPE Cray XD670,1.14.00,1.17.00,success
```

Here is an example of the update for HPE Cray XD670 BIOS from version CUXD670_5.29_v1.14 to CUXD670_5.32_v2.00.

```
$ cat config.ini

[Image]
update_image_path_xd670 = CUXD670_5.32_v2.00_signed.bin.hpm
[Target]
update_target = BIOS

##Allowed options are:
#XD670 - BMC, BMCImage2, BIOS, BIOS2, SCM_CPLD1_MB_CPLD1, BPB_CPLD, GPU_ALL
##BMC is equivalent to BMCImage1 here for Cray XD670 systems with Non-RoT motherboards (BMC
remains same as there is only a single target which is equal to BMC for RoT motherboards)
##BPB_CPLD is the update of BPB_CPLD1 and BPB_CPLD2 firmwares back to back, it requires only one
hpm file
##SCM_CPLD1_MB_CPLD1 is the update of SCM_CPLD1 and MB_CPLD1 firmwares back to back, update files
for the same should be mentioned as


#SCM_CPLD1_file.hpm MB_CPLD1_file.hpm ##first file to be of SCM_CPLD1 and space and followed by
MB_CPL1 hpm file
##GPU_ALL is to update all components in the GPU (Firmware type should be HMC)

##all targets are case-sensitive. follow the exact same cases.
```

```
[Options]
power_state = NA
#NA on off
##applicable only for Cray XD670


[Firmware_type]
update_image_type =
##Allowed options(Default: HPM):
#Mention firmware type only for below components
#HPM : For BMC, BIOS and CPLD components(MB_CPLD, SCM_CPLD, BPB_CPLD)
#HMC : For complete GPU update


$ ansible-playbook system_firmware_update.yml -i inventory -e @system_credentials.yml

PLAY [version 1.0 System Firmware Update for HPE Cray XD670 model systems]
*********************************************************************

TASK [Gathering Facts]
*************************************************************************************************
*****************************************************************************
ok: [10.xx.xx.xx]

TASK [System Firmware Update Status result will be uploaded to the below csv file]
*************************************************************************************************
******************************
ok: [10.xx.xx.xx]

TASK [Running Firmware Update for Cray XD Servers]
*************************************************************************************************
********************************************************
ok: [10.xx.xx.xx]

TASK [Writing Firmware Upgrade status details to System_FW_Update_2024-08-01_2:05:40.csv file]
*************************************************************************************************
*
changed: [10.xx.xx.xx]

PLAY RECAP
*************************************************************************************************
******************************************************************************
10.xx.xx.xx                    : ok=4    changed=1    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0
```

Following is an example output of a CSV file:

```
$ cat System_FW_Update_2024-08-01_2:05:40.csv

IP_Address,Model,BIOS_Pre_Ver,BIOS_Post_Ver,Update_Status
10.xx.xx.xx,HPE Cray XD670,CUXD670_5.29_v1.14,CUXD670_5.32_v2.00,success
```

Here is an example of the update for CRAY XD670 BPB_CPLD

```
$ cat config.ini

[Image]
update_image_path_xd670 = G593_SD0_20221228_R05R32_0xDF82.rcu

[Target]
update_target = BPB_CPLD
```

```
##Allowed options are:
#XD670 - BMC, BMCImage2, BIOS, BIOS2, SCM_CPLD1_MB_CPLD1, BPB_CPLD, GPU_ALL
##BMC is equivalent to BMCImage1 here for Cray XD670 systems with Non-RoT motherboards (BMC
remains same as there is only a single target which is equal to BMC for RoT motherboards)
##BPB_CPLD is the update of BPB_CPLD1 and BPB_CPLD2 firmwares back to back, it requires only one
hpm file

##SCM_CPLD1_MB_CPLD1 is the update of SCM_CPLD1 and MB_CPLD1 firmwares back to back, update files
for the same should be mentioned as
#SCM_CPLD1_file.hpm MB_CPLD1_file.hpm ##first file to be of SC_CPLD1 and space and followed by
MB_CPL1 hpm file
##GPU_ALL is to update all components in the GPU (Firmware type should be HMC)
##all targets are case-sensitive. follow the exact same cases.


[Options]
power_state = NA
#NA on off
##applicable only for Cray XD670

[Firmware_type]
update_image_type =
##Allowed options(Default: HPM):
#Mention firmware type only for below components
#HPM : For BMC, BIOS and CPLD components(MB_CPLD, SCM_CPLD, BPB_CPLD)
#HMC : For complete GPU update

$ ansible-playbook system_firmware_update.yml -i inventory -e @system_credentials.yml


PLAY [version 1.0 System Firmware Update for HPE Cray XD670 model systems]
*************************************************************************

TASK [Gathering Facts]
*******************************************************************************************************
*********************************************************************************
ok: [10.xx.xx.xx]

TASK [System Firmware Update Status result will be uploaded to the below csv file]
*******************************************************************************************************
***********************
ok: [10.xx.xx.xx]

TASK [Running Firmware Update for Cray XD Servers]
*******************************************************************************************************
*********************************************************
ok: [10.xx.xx.xx]

TASK [Writing Firmware Upgrade status details to System_FW_Update_2023-12-14_23:47:32.csv file]
*******************************************************************************************************
**********
changed: [10.xx.xx.xx]

PLAY RECAP
*******************************************************************************************************
**********************************************************************************************
10.xx.xx.xx                  : ok=4    changed=1    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0

$ cat System_FW_Update_2023-12-14_23:47:32.csv
```

```
IP_Address,Model,Update_Status,Remarks
10.xx.xx.xx,HPE Cray XD670,Success, plug out and plug in power cables physically
```

Note: After the update, plug out and plug in the setup physically

Here is an example of the update for CRAY XD670 SCM_CPLD and MB_CPLD

```
$ cat config.ini
[Image]
update_image_path_xd670= SCM_CDCR112_20230222_R07_0x4DD0.rcu EGS_20230309_R16_0x1A4C.rcu

[Target]
update_target = SCM_CPLD1_MB_CPLD1
##Allowed options are:
#XD670 - BMC, BMCImage2, BIOS, BIOS2, SCM_CPLD1_MB_CPLD1, BPB_CPLD, GPU_ALL
##BMC is equivalent to BMCImage1 here for Cray XD670 systems with Non-RoT motherboards (BMC
remains same as there is only a single target which is equal to BMC for RoT motherboards)
##BPB_CPLD is the update of BPB_CPLD1 and BPB_CPLD2 firmwares back to back, it requires only one
hpm file
##SCM_CPLD1_MB_CPLD1 is the update of SCM_CPLD1 and MB_CPLD1 firmwares back to back, update files
for the same should be mentioned as
#SCM_CPLD1_file.hpm MB_CPLD1_file.hpm ##first file to be of SC_CPLD1 and space and followed by
MB_CPL1 hpm file
##GPU_ALL is to update all components in the GPU (Firmware type should be HMC)
##all targets are case-sensitive. follow the exact same cases.

[Options]
power_state = on
#NA on off
##applicable only for Cray XD670


[Firmware_type]
update_image_type =
##Allowed options(Default: HPM):
#Mention firmware type only for below components
#HPM : For BMC, BIOS and CPLD components(MB_CPLD, SCM_CPLD, BPB_CPLD)
#HMC : For complete GPU update

$ ansible-playbook power_state_XD670.yml -i inventory -e @system_credentials.yml

PLAY [version 1.0 System Firmware Update for HPE Cray XD670 model systems]
**************************************************************************

TASK [Gathering Facts]
*********************************************************************************************
***********************************************************************************
ok: [10.xx.xx.xx]

TASK [Power states of the nodes will be uploaded to the below csv file]
*********************************************************************************************
**********************************
ok: [10.xx.xx.xx]

TASK [Getting Power State of Cray XD670 Server nodes]
*********************************************************************************************
****************************************************
ok: [10.xx.xx.xx]
```

```
TASK [Writing Power status details to Power_State_CrayXD670_2023-12-15_00:03:49.csv file]
*************************************************************************************************
*****
changed: [10.xx.xx.xx]

PLAY RECAP
*************************************************************************************************
*****************************************************************************************

10.xx.xx.xx                : ok=4    changed=1    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0

$ cat Power_State_CrayXD670_2023-12-15_00:03:49.csv

IP_Address,Model,Power_State

10.xx.xx.xx,HPE Cray XD670,On


$ ansible-playbook system_firmware_update.yml -i inventory -e @system_credentials.yml


PLAY [version 1.0 System Firmware Update for HPE Cray XD670 model systems]
****************************************************************************

TASK [Gathering Facts]
*************************************************************************************************
*****************************************************************************
ok: [10.xx.xx.xx]

TASK [System Firmware Update Status result will be uploaded to the below csv file]
*************************************************************************************************
**********************
ok: [10.xx.xx.xx]

TASK [Running Firmware Update for Cray XD Servers]
*************************************************************************************************
*****************************************
[WARNING]: Collection community.general does not support Ansible version 2.10.8
ok: [10.xx.xx.xx]

TASK [Writing Firmware Upgrade status details to System_FW_Update_2023-12-15_00:41:10.csv file]
*************************************************************************************************
**********
changed: [10.xx.xx.xx]

PLAY RECAP
*************************************************************************************************
******************************************************************************************
10.xx.xx.xx                : ok=4    changed=1    unreachable=0    failed=0    skipped=0
rescued=0    ignored=0

$ cat System_FW_Update_2023-12-15_00:41:10.csv

IP_Address,Model,Update_Status,Remarks
10.xx.xx.xx,HPE Cray XD670,Success, plug out and plug in power cables physically
```

Note: After the update, plug out and plug in the setup physically.

GPU update

```
$cat config.ini
[Image]
update_image_path_xd670 = nvfw_HGX-H100x8_0002_231101.5.0_prod-signed.fwpkg

[Target]
update_target = GPU_ALL
##Allowed options are:
#XD670 - BMC, BMCImage2, BIOS, BIOS2, SCM_CPLD1_MB_CPLD1, BPB_CPLD, GPU_ALL
##BMC is equivalent to BMCImage1 here for Cray XD670 systems with Non-RoT motherboards (BMC
remains same as there is only a single target which is equal to BMC for RoT motherboards)

##BPB_CPLD is the update of BPB_CPLD1 and BPB_CPLD2 firmwares back to back, it requires only one
hpm file
##SCM_CPLD1_MB_CPLD1 is the update of SCM_CPLD1 and MB_CPLD1 firmwares back to back, update files
for the same should be mentioned as
#SCM_CPLD1_file.hpm MB_CPLD1_file.hpm ##first file to be of SC_CPLD1 and space and followed by
MB_CPL1 hpm file
##GPU_ALL is to update all components in the GPU (Firmware type should be HMC)
##all targets are case-sensitive. follow the exact same cases.

[Options]
power_state = NA
##Allowed options are: NA on off
##applicable only for XD670


[Firmware_type]
update_image_type =
##Allowed options(Default: HPM):
#Mention firmware type only for below components
#HPM : For BMC, BIOS and CPLD components(MB_CPLD, SCM_CPLD, BPB_CPLD)
#HMC : For complete GPU update

$ ansible-playbook -i inventory system_firmware_update.yml -e @system_credentials.yml

PLAY [version 1.0 System Firmware Update for HPE Cray XD670 model systems]
*******************************************

TASK [Gathering Facts]
*************************************************************************************************
********************************************************************
ok: [10.xx.xx.xx]

TASK [System Firmware Update Status result will be uploaded to the below csv file]
*************************************************************************************************
********
ok: [10.xx.xx.xx]

TASK [Running Firmware Update for Cray XD Servers]
*************************************************************************************************
****************************************
[WARNING]: Collection community.general does not support Ansible version 2.10.8
ok: [10.xx.xx.xx]

TASK [Writing Firmware Upgrade status details to System_FW_Update_2024-06-18_02:19:46.csv file]
*************************************************************************************************
changed: [10.xx.xx.xx]
```

```
PLAY RECAP
************************************************************************************************
***************************************************************************
10.xx.xx.xx              : ok=4    changed=1   unreachable=0    failed=0    skipped=0
rescued=0    ignored=0

$ cat System_FW_Update_2024-06-18_02\:19\:46.csv

IP_Address,Model,Update_Status,Remarks
10.xx.xx.xx,HPE Cray XD670 RoT,Update Triggered , It will take nearly 30 to 40 minutes to update
baseboard firmware. The target system will be reboot once the firmware update procedure completed
```

Documentation feedback

Hewlett Packard Enterprise is committed to providing documentation that meets your needs. To help us improve the documentation, use the **Feedback** button and icons (located at the bottom of an opened document) on the Hewlett Packard Enterprise Support Centre portal (**https://www.hpe.com/support/hpesc**) to send any errors, suggestions, or comments. All document information is captured by the process.