



NVIDIA DGX GB200 Firmware Update Guide

NVIDIA Corporation

Oct 15, 2025

Rack Firmware Updates

1 Overview	3
1.1 Compute Tray Firmware Update Process	4
1.1.1 Method 1 - BCM/NVIDIA Mission Control Integrated Firmware Update for Compute Tray	5
1.1.2 Method 2 - Stand Alone nvfwupd Tool for Compute Tray	10
1.1.2.1 Applying and Verifying Firmware Update Success	15
1.2 NVLink Switch Firmware Update Process	15
1.2.1 NVLink Switch Tray Assumptions	16
1.2.2 Method 1 - BCM/NVIDIA Mission Control FW Update Integrated Process for NVLink Switch	18
1.2.3 Method 2 - Stand Alone nvfwupd Tool FW Update Process for NVLink Switch	20
1.2.4 Method 3 - FW Updates within NVOS for NVLink Switch	23
1.3 NVLink Switch - Updating NVOS	24
2 Power Shelf Firmware Update Process	29
3 Bluefield and CX7 Firmware Update Process	33
3.1 Combined CX-7 and Bluefield Update	34
4 Software Update	35
5 Operational Security Requirements	37
5.1 Factory Reset after Debug Token Usage	37
6 Additional Information	39
6.1 Compute Tray BMC Username	39

The *NVIDIA DGX GB200 Firmware Update Guide* is also available as a [PDF](#).

Chapter 1. Overview

Overall, firmware updates using the NVIDIA Base Command Manager (BCM) 11 software for a GB200 NVL72 rack can be done once all the GB200 compute trays, NVLink switch trays, and power shelves are up in BCM. The latest FW/SW recipe must be followed for the installation on both devices to be successful.

The processes described in this guide apply to DGX GB200 NVL72 software version 1.2.2 and later.

Note

FW packages for DGX SuperPOD GB200 are unique and different from the reference GB200 architecture package.

Table 1: Reference: DGX GB200 Compute Tray Files Required for Update on DGX SuperPOD (As of BCM 11 1.2 GA)

Component	DGX Recipe Version	FW Version	Filename
DGX GB200 SW/FW Release Notes	1.3		
Compute BMC bundle			nvfw_DGX-GBX00_0023_<date>.*_custom_prod-signed.fwpkg
Compute HMC bundle			nvfw_HGX-GBX00_0023_<date>.*_custom_prod-signed.fwpkg
BF3			fw-Bluefield-3-rel-*.bin
CX7			fw-ConnectX7-rel-*.bin
Switch NVOS			nvos-amd64-*.bin
Switch BMC bundle			nvfw_GB200-P4978_0004.*.fwpkg
Switch BIOS bundle			nvfw_GB200-P4978_0006.*.fwpkg
Switch CPLD bundle			nvfw_GB200-P4978_0007.*.fwpkg
Powershell PSU			NVIDIA_5500_APP_*.tar
Powershell PMC			common-pmc-3.*tar

Firmware updates for the GB200 compute trays can be done by:

1. BCM 11 integrated firmware update tool
2. Standalone nvfwupd tool

GB200 Compute Tray Firmware Update - General Steps

1. Obtain the compute tray package
2. Ensure that compute tray BMC has username “admin” enabled and that the credentials are known. If username “admin” does not exist or is disabled, it must be created and enabled before the compute tray update. BCM or any rack management systems should migrate to using “admin” as default BMC account going forward as the previously used “root” will be disabled going forward. Please see Appendix A.1 before proceeding with the Update.
3. If using BCM to do the firmware update
 1. Place the files in `/cm/local/apps/cmd/etc/htdocs/bios/firmware/gb200`
 2. Confirm that in the NVLink Switch bmcsettings, the firmware management mode is set to GB200
 3. Check the current nodes FW versions against the update packages
 4. Do a dry run to confirm the FW will update to the expected versions
4. Update the BMC package first (Compute BMC bundle), then the compute tray package (Compute HMC bundle). AUX power cycle the trays after each component update is complete

NVLink Switch Tray Firmware Update - General Steps

1. Obtain the NVLink Switch firmware
2. If using BCM to do the firmware update
 1. Place the files in `/cm/local/apps/cmd/etc/htdocs/bios/firmware/gb200sw`
 2. Confirm that in the NVLink Switch bmcsettings, the firmware management mode is set to GB200sw
3. Check the current NVLink Switch FW versions against the update packages
4. Do a dry run to confirm the FW will update to the expected versions
5. Update the tray level firmware first in this order
 1. BMC+FPGA+ERoT (Switch BMC bundle)
 2. CPLD1 CPLD2 CPLD3 CPLD4 (Switch CPLD bundle)
 3. SBIOS+EROT (Switch BIOS bundle)
6. Update the NVOS from within the OS or use ZTP. (Switch NVOS)
7. Aux power cycle the trays after each component update is complete.

1.1. Compute Tray Firmware Update Process

1.1.1. Method 1 - BCM/NVIDIA Mission Control Integrated Firmware Update for Compute Tray

To use the firmware update tool in BCM 11 an NVIDIA Mission Control enabled license must be registered.

1. Place Firmware Update Packages in the Correct BCM Directory
`/cm/local/apps/cmd/etc/htdocs/bios/firmware/gb200`
2. Copy the prod-signed.fwpkg images up to the BCM head node. The files must be placed in the following directory to be visible to the 'firmware' command:

```
scp <binary files> user@<headnode>:/cm/local/apps/cmd/etc/htdocs/bios/
→firmware/gb200
```

Reference: BCM file directory structure for firmware updates.

```
/cm/local/apps/cmd/etc/htdocs/bios/firmware/

README.md b200/ gb200/ gb200sw/ gh200/ h100/ ilo/

# The gb200 folder is for compute tray firmware, the gb200sw folder
→is
# for NVLink Switch firmware
```

1. Use the firmware info command in BCM to gather information on the current FW levels of the nodes. It will detail the files and what their purpose is.
2. Use the firmware info command in BCM to gather information on the current FW levels of the nodes. It will detail the files and what their purpose is.

```
cmsh;device;firmware info

[T06-HEAD-01->device]% firmware info

Device      Filename
→Component  Version      State      Progress
→Result    Size        Date
-----
→-----
→-----
T06-HEAD-01  nvfw_DGX-GBX00_0024_250215.1.0_custom_prod-signed.fwpkg
→GB200-BMC   DGX-GBX00_0024_250215.1.0_custom available  N/A      64MiB
→ 2025-02-15, 16:39:41
T06-HEAD-01  nvfw_GB200-P4978_0004_250213.1.0_prod-signed.fwpkg
→GB200-Switch GB200-P4978_0004_250213.1.0  available  N/A      75MiB
→2025-02-13, 10:23:28
T06-HEAD-01  nvfw_GB200-P4978_0006_250205.1.0_prod-signed.fwpkg
→GB200-Switch GB200-P4978_0006_250205.1.0  available  N/A      16.2MiB
→2025-02-05, 15:11:49
T06-HEAD-01  nvfw_GB200-P4978_0007_250121.1.2_custom_prod-signed.fwpkg
→GB200-Switch GB200-P4978_0007_250121.1.2_custom available  N/A      1.
→64MiB 2025-01-21, 13:55:30
T06-HEAD-01  nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.fwpkg
```

(continues on next page)

(continued from previous page)

```
→GB200-Compute HGX-GBX00_0023_250223.1.1_custom available N/A
→114MiB 2025-02-23, 20:20:42
```

Note: This will display the file names and target (such as GB200 or Switch) of all available firmware binaries. If the files do not show up with this command, they cannot be flashed by the update tool. The officially released packages will have a common filename structure starting with nvfw_DGX-GBX00_<identifier>_<date>.

3. Confirm GB200 Tray BMC Access/Connectivity.

1. The BMC of each node needs to be configured in BCM. This should be done at the category level. Ensure that no bmcsettings are added at the node level so that the compute trays inherit the settings from the category level.
2. Enter cmsh and show the current BMC settings for a given node or use the category level for GB200 compute trays since all of their default passwords are the same (for DGX).

```
#category level
category; use <dgx-category>;bmcsettings; show

#device level
device; use <device name>; bmcsettings; show
```

Only use the device level to confirm that nothing has been set.

It will show as if they have not been set before as indicated by an asterisk.

```
[bcm11-headnode->device*[a08-p1-dgx-04-c18\*]->bmcsettings\*]%

#use this command to clear uncommitted changes
refresh
```

3. Populate the bmcsettings fields in the dgx-gb200 category if it is not already populated.

```
cmsh;category use dgx-gb200;bmcsettings;
set username admin
set password <Password of choice>
set userid 1
set firmwaremanagemode gb200
commit
```

Note: It is critical that the firmware management mode here is set to gb200.

4. Test that the BMC is configured by reading the current FW versions.

```
#at the device level
cmsh; device use <dgx-node-name>; firmware status

[maple->device[dgx-gb200-m07-c1]]% firmware status

Device Filename Component Version State Progress Result Size Date
-----
dgx-gb200-m07-c1 CX7_0 28.42.1270 current N/A N/A
dgx-gb200-m07-c1 CX7_1 28.42.1270 current N/A N/A
dgx-gb200-m07-c1 CX7_2 28.42.1270 current N/A N/A
```

(continues on next page)

(continued from previous page)

```

dgx-gb200-m07-c1 CX7_3 28.42.1270 current N/A N/A
dgx-gb200-m07-c1 FW_BMC_0 GB200Nv1-24.12-8 current N/A N/A
dgx-gb200-m07-c1 FW_CPLD_0 0x00 0x0b 0x03 0x04 current N/A N/A
dgx-gb200-m07-c1 FW_CPLD_1 0x00 0x0b 0x03 0x04 current N/A N/A
dgx-gb200-m07-c1 FW_CPLD_2 0x00 0x10 0x01 0x0f current N/A N/A
dgx-gb200-m07-c1 FW_CPLD_3 0x00 0x10 0x01 0x0f current N/A N/A
dgx-gb200-m07-c1 FW_ERoT_BMC_0 01.03.0262.0000_n04 current N/A N/A
dgx-gb200-m07-c1 Full_FW_Image_NIC_Slot_4 32.42.1000 current N/A N/A
dgx-gb200-m07-c1 Full_FW_Image_NIC_Slot_7 32.42.1000 current N/A N/A
dgx-gb200-m07-c1 UEFI buildbrain-gcid-38635631 current N/A N/A

```

#At the category level to see all of the compute tray FW in one shot
cmsh; device;firmware -c dgx-gb200 status

#At the rack level
cmsh; device;firmware -r <rack location> status

4. As a validation step prior to executing the flash, a dry-run command is supported to show exactly what will be changing when the firmware is applied:

1. Perform a dry run of the BMC FW

```

cmsh;device; firmware flash nvfw_DGX-GBX00_0023_241223.1.0_custom_
↪prod-signed.fwpkg --dry-run -n <device name>

```

The <device name> can have some regex to apply the change to multiple devices simultaneously:

- ▶ dgx-gb200-r1-c[1-2] - This will run the command against both dgx-gb200-r1-c1 and dgx-gb200-r1-c2
- ▶ Device names can also be comma separated to run against multiple individual devices: dgx-gb200-r1-c1, dgx-gb200-r1-c2

Example: Dry run output

```

s03-p1-dgx-01-c06 HGX_FW_BMC_0 HGX_FW_BMC_0 GB200Nv1-25.01-D GB200Nv1-
↪25.01-E no install good
s03-p1-dgx-01-c06 HGX_FW_CPLD_0 HGX_FW_CPLD_0 0.1C 0.1C yes skip good
s03-p1-dgx-01-c06 HGX_FW_CPU_0 HGX_FW_CPU_0 02.03.19 02.03.20 no
↪install good
s03-p1-dgx-01-c06 HGX_FW_CPU_1 HGX_FW_CPU_1 02.03.19 02.03.20 no
↪install good
s03-p1-dgx-01-c06 HGX_FW_ERoT_BMC_0 HGX_FW_ERoT_BMC_0 01.04.0008.0000_
↪n04 01.04.0008.0000_n04 yes skip good
s03-p1-dgx-01-c06 HGX_FW_ERoT_CPU_0 HGX_FW_ERoT_CPU_0 01.04.0008.0000_
↪n04 01.04.0008.0000_n04 yes skip good
s03-p1-dgx-01-c06 HGX_FW_ERoT_CPU_1 HGX_FW_ERoT_CPU_1 01.04.0008.0000_
↪n04 01.04.0008.0000_n04 yes skip good
s03-p1-dgx-01-c06 HGX_FW_ERoT_FPGA_0 HGX_FW_ERoT_FPGA_0 01.04.0008.
↪0000_n04 01.04.0008.0000_n04 yes skip good
s03-p1-dgx-01-c06 HGX_FW_ERoT_FPGA_1 HGX_FW_ERoT_FPGA_1 01.04.0008.
↪0000_n04 01.04.0008.0000_n04 yes skip good
s03-p1-dgx-01-c06 HGX_FW_FPGA_0 HGX_FW_FPGA_0 1.20 1.20 yes skip good

```

(continues on next page)

(continued from previous page)

```
s03-p1-dgx-01-c06 HGX_FW_FPGA_1 HGX_FW_FPGA_1 1.20 1.20 yes skip good
s03-p1-dgx-01-c06 HGX_FW_GPU_0 HGX_FW_GPU_0 97.00.82.00.13 97.00.82.
  ↳00.19 no install good
s03-p1-dgx-01-c06 HGX_FW_GPU_1 HGX_FW_GPU_1 97.00.82.00.13 97.00.82.
  ↳00.19 no install good
s03-p1-dgx-01-c06 HGX_FW_GPU_2 HGX_FW_GPU_2 97.00.82.00.13 97.00.82.
  ↳00.19 no install good
s03-p1-dgx-01-c06 HGX_FW_GPU_3 HGX_FW_GPU_3 97.00.82.00.13 97.00.82.
  ↳00.19 no install good
```

2. Ensure that the values that are going to be updated are the expected versions.

5. Start the firmware update.

```
cmsh -c 'device; firmware flash nvfw_DGX-GBX00_0023_250614.1.0_custom_
  ↳prod-signed.fwpkg -n <device name>'
```

6. Once the payload is uploaded to the node it will say good.

```
[T06-HEAD-01->device]% firmware flash nvfw_DGX-GBX00_0023_250614.1.0_
  ↳custom_prod-signed.fwpkg -n s03-p1-dgx-01-c{04..06}
```

Device	flashing file
↳Result	
-----	-----
↳-----	
s03-p1-dgx-01-c04	nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
↳fwpkg good	
s03-p1-dgx-01-c05	nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
↳fwpkg good	
s03-p1-dgx-01-c06	nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
↳fwpkg good	

7. When the command completes, check the status of the update until it has completed. This will have a percentage complete while the flashing is ongoing and a complete message when the flash has finished.

```
cmsh -c 'device; firmware status -n <device name>'
```

```
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
  ↳fwpkg HGX_FW_BMC_0 GB200Nv1-25.01-D flashing 0.0% 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
  ↳fwpkg HGX_FW_CPU_0 02.03.19 flashing 0.0% 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
  ↳fwpkg HGX_FW_CPU_1 02.03.19 flashing 0.0% 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
  ↳fwpkg HGX_FW_GPU_0 97.00.82.00.13 flashing 0.0% 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
  ↳fwpkg HGX_FW_GPU_1 97.00.82.00.13 flashing 0.0% 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
  ↳fwpkg HGX_FW_GPU_2 97.00.82.00.13 flashing 0.0% 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
  ↳fwpkg HGX_FW_GPU_3 97.00.82.00.13 flashing 0.0%
```

(continues on next page)

(continued from previous page)

```

s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
→fwpkg HGX_FW_BMC_0 GB200Nv1-25.01-D -> GB200Nv1-25.01-E pending N/A
→success: medium-specific reset or dc power cycle or ac power cy+ 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
→fwpkg HGX_FW_CPU_0 02.03.19 -> 02.03.20 pending N/A success: medium-
→specific reset or dc power cycle or ac power cy+ 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
→fwpkg HGX_FW_CPU_1 02.03.19 -> 02.03.20 pending N/A success: medium-
→specific reset or dc power cycle or ac power cy+ 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
→fwpkg HGX_FW_GPU_0 97.00.82.00.13 -> 97.00.82.00.19 pending N/A
→success: medium-specific reset or dc power cycle or ac power cy+ 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
→fwpkg HGX_FW_GPU_1 97.00.82.00.13 -> 97.00.82.00.19 pending N/A
→success: medium-specific reset or dc power cycle or ac power cy+ 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
→fwpkg HGX_FW_GPU_2 97.00.82.00.13 -> 97.00.82.00.19 pending N/A
→success: medium-specific reset or dc power cycle or ac power cy+ 114MiB
s03-p1-dgx-01-c06 nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.
→fwpkg HGX_FW_GPU_3 97.00.82.00.13 -> 97.00.82.00.19 pending N/A
→success: medium-specific reset or dc power cycle or ac power cy+ 114MiB

```

At the end of the BMC update, the administrator can AC Cycle the GB200 node(s) to **complete** the BMC update, **then** proceed with updating other components.

****Note:**** It is important to AC/AUX cycle the target host after the CPLD and BMC updates because the BMC has limited memory and cannot store another firmware package. AC cycling clears the memory and applies changes, allowing the HMC update to proceed successfully.

8. Do the AC Cycle after each .fwpkg completed the firmware update

Power Cycle Method - through the AUX_PWR_CYCLE - Redfish

#or use cmsh to power off the node

```
cmsh;device;use <compute node under test>;power off
```

#or to do multiples

```
cmsh;device;foreach -c dgx-gb200 (power off)
```

#Do this next to effectively AC Power cycle (removal of auxiliary power)

```
curl -k -u "${USER}:${PASS}" -H "Content-Type: application/json" -X POST \
  -d '{"ResetType":"AuxPowerCycle"}' \
  https://<rf0 ip>/redfish/v1/Chassis/BMC_0/Actions/Oem/NvidiaChassis.
→AuxPowerReset
```

#use redfish to power on

#or use cmsh to power on the node

```
cmsh;device;use <compute node under test>;power on
```

(continues on next page)

(continued from previous page)

```
#or to do multiples
cmsh;device;foreach -c dgx-gb200 (power on)

#or
cmsh;device;power on -c dgx-gb200 #this does all nodes in the category
cmsh;device;power on -n <specific nodes>
```

9. If issues arise, getting the debug output can help root cause some issues. Use the flash command with debug options enabled to get debug output

```
cmsh -c 'device; firmware flash nvfw_DGX-GBX00_0023_241223.1.0_custom_
→prod-signed.fwpkg -n <device name> -v --debug'
```

1.1.2. Method 2 - Stand Alone nvfwupd Tool for Compute Tray

If the license does not support NVIDIA Mission Control, the built in cm-nvfwupd will not work. Download the latest standalone nvfwupd tool from Enterprise Portal - v2.0.7 or later: [Announcement: nvfwupd tool version 2.0.7](#) This tool or method is used independent of BCM.

NOTE: These instructions only cover the update of a single compute tray. The stand-alone tool supports simultaneous upgrades for multiple systems, and multiple components like the compute trays and NVLink switches together. Please refer to Chapters 17, 18 and 19 of the *NVIDIA Firmware Update Guide* that is included with the nvfwupdate tool.

Get the correct FW update packages for update. To see the full contents of a fwupd.pkg, use the show_pkg_content command.

```
./nvfwupd show_pkg_content -p
./nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.fwpkg
```

Get current state of the hardware with show_version.

```
./nvfwupd -t ip=<rf0 ip> user=${USER} password=${PASS} \
  servertype=GB200 show_version -p ./nvfw_GB200-P4972_0012_250214.1.0_
→custom_prod-signed.fwpkg \
  ./nvfw_GB200-P4975_0011_250206.1.1_custom_recovery_prod-signed.fwpkg
```

System Model: GB200 NVL

Part number: 699-24764-0001-RC1

Serial number: 1334524170073

Packages: ['GB200-P4972_0012_250214.1.0_custom', 'GB200-P4975_0011_250206.1.1_
→custom_recovery']

Connection Status: Successful

Firmware Devices:

(continues on next page)

(continued from previous page)

AP Name →To-Date	Sys Version	Pkg Version	Up-
→-----			
CX7_0	28.43.2108	N/A	No
CX7_1	28.43.2108	N/A	No
CX7_2	28.43.2108	N/A	No
CX7_3	28.43.2108	N/A	No
FW_BMC_0	GB200Nv1-25.01-D	GB200Nv1-25.01-E	No
FW_CPLD_0	0x00 0x0b 0x03 0x04	N/A	No
FW_CPLD_1	0x00 0x0b 0x03 0x04	N/A	No
FW_CPLD_2	0x00 0x10 0x01 0x0f	N/A	No
FW_CPLD_3	0x00 0x10 0x01 0x0f	N/A	No
FW_ERoT_BMC_0	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
Full_FW_Image_NIC_Slot_4	32.43.2408	N/A	No
Full_FW_Image_NIC_Slot_7	32.43.2408	N/A	No
UEFI	buildbrain-gcid-39281046	N/A	No
HGX_FW_BMC_0	GB200Nv1-25.01-D	N/A	No
HGX_FW_CPLD_0	0.1C	N/A	No
HGX_FW_CPU_0	02.03.19	N/A	No
HGX_FW_CPU_1	02.03.19	N/A	No
HGX_FW_ERoT_BMC_0	01.04.0008.0000_n04	01.03.0196.0001	Yes
HGX_FW_ERoT_CPU_0	01.04.0008.0000_n04	01.03.0196.0001	Yes
HGX_FW_ERoT_CPU_1	01.04.0008.0000_n04	01.03.0196.0001	Yes
HGX_FW_ERoT_FPGA_0	01.04.0008.0000_n04	01.03.0196.0001	Yes
HGX_FW_ERoT_FPGA_1	01.04.0008.0000_n04	01.03.0196.0001	Yes
HGX_FW_FPGA_0	1.20	N/A	No
HGX_FW_FPGA_1	1.20	N/A	No
HGX_FW_GPU_0	97.00.82.00.13	1.0.61.0	No
HGX_FW_GPU_1	97.00.82.00.13	1.0.61.0	No
HGX_FW_GPU_2	97.00.82.00.13	1.0.61.0	No
HGX_FW_GPU_3	97.00.82.00.13	1.0.61.0	No
HGX_InfoROM_GPU_0	G548.0201.00.06	N/A	No
HGX_InfoROM_GPU_1	G548.0201.00.06	N/A	No
HGX_InfoROM_GPU_2	G548.0201.00.06	N/A	No
HGX_InfoROM_GPU_3	G548.0201.00.06	N/A	No
HGX_PCISwitchConfig_0	01151024	N/A	No
→-----			
Error Code: 0			

Create payload .jsons for the bmc and the compute tray

Reference: UpdateBMC.json

```
{
  "Targets": []
}
```

Reference: UpdateCompute.json

```
{
```

(continues on next page)

(continued from previous page)

```
"Targets": [ "/redfish/v1/Chassis/HGX_Chassis_0" ]
}
```

Run the BMC update first.

```
./nvfwupd -t ip=<rf0 ip> user=$USER password=$PASSWORD servertime=GB200
update_fw -s BMC_Full.json -p
./nvfw_DGX-GBX00_0024_250215.1.0_custom_prod-signed.fwpkg
```

Power off the system, then do an AC Cycle.

```
./nvfwupd -t ip=<rf0 ip> user=${USER} password=${PASS} servertime=GB200
activate_fw -c PWR_OFF
```

wait 15 seconds

```
./nvfwupd -t ip=<rf0 ip> user=${USER} password=${PASS} servertime=GB200
activate_fw -c RF_AUX_PWR_CYCLE
```

Check if the BMC update was successful.

Reference: Successful BMC update.

```
./nvfwupd -t ip=<rf0 ip> user=${USER} password=${PASS} servertime=GB200
→update_fw -s ip=<rf0 ip> user=${USER} password=${PASS} servertime=GB200
→update_fw -s BMC_Full.json -p ./nvfw_DGX-GBX00_0024_250215.1.0_custom_prod-
→signed.fwpkg
```

Updating ip address: ip=XXXX

FW package:

```
['./nvfw_DGX-GBX00_0024_250215.1.0_custom_prod-signed.fwpkg']
```

Updating ip address: ip=XXXX

FW package:

```
['./nvfw_DGX-GBX00_0024_250215.1.0_custom_prod-signed.fwpkg']
```

Ok to proceed with firmware update? <Y/N>

y

FW package:

```
['./nvfw_DGX-GBX00_0024_250215.1.0_custom_prod-signed.fwpkg']
```

Ok to proceed with firmware update? <Y/N>

y

```
{"@odata.id": "/redfish/v1/TaskService/Tasks/3", "@odata.type":
"#Task.v1_4_3.Task", "Id": "3", "TaskState": "Running", "TaskStatus":
"OK"}
```

(continues on next page)

(continued from previous page)

```
FW update started, Task Id: 3
```

```
Wait for Firmware Update to Start...
```

```
TaskState: Running
```

```
PercentComplete: 20
```

```
TaskStatus: OK
```

```
TaskState: Running
```

```
PercentComplete: 40
```

```
TaskStatus: OK
```

```
TaskState: Running
```

```
PercentComplete: 60
```

```
TaskStatus: OK
```

```
TaskState: Completed
```

```
PercentComplete: 100
```

```
TaskStatus: OK
```

```
Firmware update successful!
```

```
Overall Time Taken: 0:13:01
```

Refer to 'NVIDIA Firmware Update Document' on activation steps for new firmware to take effect.

Do the full compute tray flash (HGX). Ensure that the system is fully up and, in its OS, to be able to do the GPU VBIOS updates.

```
./nvfwupd -t ip=<rf0 ip> user=${USER} password=${PASS} servertime=GB200  
→update_fw -s Compute_Full.json -p ./nvfw_HGX-GBX00_0023_250223.1.1_custom_  
→prod-signed.fwpkg
```

Like the BMC, power down the system and then do an AUX power cycle.

Power on the machine, let it provision/boot up, then check the firmware level again

```
./nvfwupd -t ip=<rf0 ip> user=${USER} password=${PASS} servertime=GB200 show_  
→version -p ./nvfw_HGX-GBX00_0023_250223.1.1_custom_prod-signed.fwpkg ip=10.  
→78.194.13 user=root password=openBmc servertime=GB200 show_version -p ./  
→nvfw_DGX-GBX00_0024_250215.1.0_custom_prod-signed.fwpkg ./nvfw_HGX-GBX00_  
→0023_250223.1.1_custom_prod-signed.fwpkg
```

```
System Model: GB200 NVL
```

(continues on next page)

(continued from previous page)

Part number: 692-13809-2404-RC1

Serial number: 1330125050101

Packages: ['DGX-GBX00_0024_250215.1.0_custom',
 'HGX-GBX00_0023_250223.1.1_custom']

Connection Status: Successful

Firmware Devices:

AP Name	Sys Version	Pkg Version	
↪Up-To-Date			
-----	-----	-----	----
↪-----			
CX7_0	28.43.2108	N/A	No
CX7_1	28.43.2108	N/A	No
CX7_2	28.43.2108	N/A	No
CX7_3	28.43.2108	N/A	No
FW_BMC_0	GB200Nv1-25.01-E	GB200Nv1-25.01-E	Yes
FW_CPLD_0	0x00 0x0b 0x03 0x04	N/A	No
FW_CPLD_1	0x00 0x0b 0x03 0x04	N/A	No
FW_CPLD_2	0x00 0x10 0x01 0x0f	N/A	No
FW_CPLD_3	0x00 0x10 0x01 0x0f	N/A	No
FW_ERoT_BMC_0	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
Full_FW_Image_NIC_Slot_4	32.43.2408	N/A	No
Full_FW_Image_NIC_Slot_7	32.43.2408	N/A	No
UEFI	buildbrain-gcid-39556194	N/A	No
HGX_FW_BMC_0	GB200Nv1-25.01-E	GB200Nv1-25.01-E	Yes
HGX_FW_CPLD_0	0.1C	0.1C	Yes
HGX_FW_CPU_0	02.03.20	02.03.20	Yes
HGX_FW_CPU_1	02.03.20	02.03.20	Yes
HGX_FW_ERoT_BMC_0	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
HGX_FW_ERoT_CPU_0	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
HGX_FW_ERoT_CPU_1	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
HGX_FW_ERoT_FPGA_0	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
HGX_FW_ERoT_FPGA_1	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
HGX_FW_FPGA_0	1.20	1.20	Yes
HGX_FW_FPGA_1	1.20	1.20	Yes
HGX_FW_GPU_0	97.00.82.00.19	97.00.82.00.19	Yes
HGX_FW_GPU_1	97.00.82.00.19	97.00.82.00.19	Yes
HGX_FW_GPU_2	97.00.82.00.19	97.00.82.00.19	Yes
HGX_FW_GPU_3	97.00.82.00.19	97.00.82.00.19	Yes
HGX_InfoROM_GPU_0	G548.0201.00.06	N/A	No
HGX_InfoROM_GPU_1	G548.0201.00.06	N/A	No
HGX_InfoROM_GPU_2	G548.0201.00.06	N/A	No
HGX_InfoROM_GPU_3	G548.0201.00.06	N/A	No
HGX_PCIESwitchConfig_0	01151024	N/A	No

1.1.2.1 Applying and Verifying Firmware Update Success

First connect to the GB200 tray BMC OS, then:

1. Power off the host.

```
# Checks that the current status is on
curl -k -u ${USER}:${PASS} https://${BMCIP}/redfish/v1/Systems/System_0 |
  jq '.PowerState'
```

```
# Shuts down the OS
# Graceful shutdown:
curl -k -u ${USER}:${PASS} \
  https://${BMCIP}/redfish/v1/Systems/System_0/Actions/ComputerSystem.
  Reset \
  -d '{"ResetType": "GracefulShutdown"}' -X POST

# Force power off:
curl -k -u ${USER}:${PASS} \
  https://${BMCIP}/redfish/v1/Systems/System_0/Actions/ComputerSystem.
  Reset \
  -d '{"ResetType": "ForceOff"}' -X POST
```

2. AC (AUX) cycle the node.

```
curl -k -u ${USER}:${PASS} \
  https://${BMCIP}/redfish/v1/Chassis/BMC_0/Actions/Oem/NvidiaChassis.
  AuxPowerReset \
  -d '{"ResetType": "AuxPowerCycleForce"}' -X POST
```

3. Wait for the BMC to ping again (should take 2-3 min). Once the BMC pings, bring the host back up.

```
# Checks that the current status is off (if it is 'on' no further action
  required)
curl -k -u ${USER}:${PASS} https://${BMCIP}/redfish/v1/Systems/System_0 |
  jq '.PowerState'

# Power On
curl -k -u ${USER}:${PASS} \
  https://${BMCIP}/redfish/v1/Systems/System_0/Actions/ComputerSystem.
  Reset \
  -d '{"ResetType": "On"}' -X POST
```

4. When the BMC and host are back up, validate that the firmware install was successful.

```
cmsk -c 'device; firmware status -n <device name>'
```

1.2. NVLink Switch Firmware Update Process

For the NVLink Switch, the firmware updates consist of the firmware of the switch itself and then the NVOS software itself.

1.2.1. NVLink Switch Tray Assumptions

1. Non scale out design (NVL72x1) - all NVLink ports are connected to MN-NVLink cable cartridge
2. All tray interfaces are set to receive IPs using DHCP
3. The rack inventory import process or manual entry process must be completed, and all switch entries appear in the cmsh devices list.

Example: NVLink Switch BCM switch device list

```
root@T06-HEAD-01:~# cmsh -c "device; list -t switch -f hostname:15,mac:20,
→ip:12,status:11 |grep -i nvsw "
```

```
S03-P1-NVSW-01 E0:9D:73:F0:4C:DE 10.78.195.1 [ UP ]+
S03-P1-NVSW-02 E0:9D:73:3F:EB:28 10.78.195.2 [ UP ]+
S03-P1-NVSW-03 E0:9D:73:3F:E7:30 10.78.195.3 [ UP ]+
S03-P1-NVSW-04 E0:9D:73:3F:EA:C8 10.78.195.4 [ UP ]+
S03-P1-NVSW-05 E0:9D:73:3F:E4:F0 10.78.195.5 [ UP ]+
S03-P1-NVSW-06 E0:9D:73:3F:E2:C8 10.78.195.6 [ UP ]+
S03-P1-NVSW-07 E0:9D:73:3F:E2:50 10.78.195.7 [ UP ]+
S03-P1-NVSW-08 E0:9D:73:3F:E5:18 10.78.195.8 [ UP ]+
S03-P1-NVSW-09 E0:9D:73:3F:E4:F8 10.78.195.9 [ UP ]+
S04-P1-NVSW-01 E0:9D:73:F0:41:4E 10.78.195.31 [ UP ]+
S04-P1-NVSW-02 E0:9D:73:F0:59:16 10.78.195.32 [ UP ]+
S04-P1-NVSW-03 E0:9D:73:F0:41:8E 10.78.195.33 [ UP ]+
S04-P1-NVSW-04 E0:9D:73:F0:41:36 10.78.195.34 [ UP ]+
S04-P1-NVSW-05 E0:9D:73:F0:41:A6 10.78.195.35 [ UP ]+
S04-P1-NVSW-06 E0:9D:73:F0:45:36 10.78.195.36 [ UP ]+
S04-P1-NVSW-07 E0:9D:73:F0:4D:7E 10.78.195.37 [ UP ]+
S04-P1-NVSW-08 E0:9D:73:F0:3D:56 10.78.195.38 [ UP ]+
S04-P1-NVSW-09 E0:9D:73:F0:4D:B6 10.78.195.39 [ UP ]+
```

Note: For switches, the cm-lite daemon needs to be up and running for the switch to appear as [UP]

Example: NVLink Switch BCM switch Information

```
[a03-p1-head-01->device[a05-p1-nvsw-01]]% show
```

Parameter	Value
Hostname	a05-p1-nvsw-01
IP	7.241.3.1
Network	ipminet2
Revision	
Type	Switch
Mac	E0:9D:73:3F:E0:50
Model	
Ports	0
Kind	nvlink
Control script	
Control script timeout	5
SNMP Settings	<submode>
Lowest port	1

(continues on next page)

(continued from previous page)

```

Uplinks
Disable port detection    yes
Disable port mapping     no
Activation                Sun, 23 Feb 2025 12:55:30 PST
Rack                     A05:19
Chassis                  < not set >
Access Settings          <submode>
Priority                  0
VLAN cache time          5m
Has client daemon        yes
ZTP Settings             <submode>
Subnet manager           no
Disable SNMP             yes
GUID                     00000000-0000-0000-0000-000000000000
Services                 <0 in submode>
NV configuration mode     AUTO
Members
Management network       ipminet2
Power control            rf0
Custom power script
Custom power script argument
Power distribution units
Default gateway metric   0
Switch ports
Interfaces                <3 in submode>
BMC Settings             <submode>
Userdefined1
Userdefined2
User defined resources
Supports GNSS            no
Custom ping script
Custom ping script argument
Partition base
Part number
Serial number
Notes                    <0B>
Prometheus metric forwarders <0 in submode>

```

Example: BCM NVLink Switch interfaces output

```

[a03-p1-head-01->device[B05-P1-NVSW-01]->interfaces]% list

```

Type	Network device	name	IP	Network	Start if
bmc	rf0	7.241.5.21	ipminet3	always	
physical	eth0	7.241.5.1	ipminet3	always	
physical	eth1	7.241.5.11	ipminet3	always	

4. All NVLink Switches per rack are reachable by its BMC and COMe0/COMe1 port IP address.
 1. Copper connections confirmed
 2. Speed/Bandwidth (200G for COMe0 and COMe1)

3. IP Address assigned by BCM to the COMe0 and COMe1 network (ipminetx)
5. Logical Connectivity (Access)
 1. ssh to NVLink switch BMC can be done (default user/pass = `root/JulietBmc@123`)
 2. ssh to NVOS on each NVLink switch can be done (default user/pass = `admin/Juliet1234`).

Note: If the NVLink Switch has any issues and the default NVOS password above is not working, try `admin/admin`

1.2.2. Method 1 - BCM/NVIDIA Mission Control FW Update Integrated Process for NVLink Switch

1. Get a summary of the FW update files uploaded to BCM using the `/cm/local/apps/cmd/etc/htdocs/bios/firmware/gb200sw` directory. If none exist, upload the flash files to that directory.
2. Verify the files with this command `cmsh -c 'device; firmware info'`. Ensure that all the files show up with the GB200-Switch designation.

Example: Firmware Update File List for NVLink Switches

```
cmsh;device;firmware info \ | grep -i GB200-Switch

#Or get it from the individual node entry

[a03-p1-head-01->device[a05-p1-nvsw-09]]% firmware info
```

Device	Filename	State	Progress	Result	Size	Component
→ Version						Date
-----	-----	-----	-----	-----	-----	-----
→ -----						
a03-p1-head-01	nvfw_GB200-P4978_0000_250213.1.0_dbg-signed.fwpkg	GB200-Switch	GB200-P4978_250213.1.0	available	N/A	71MiB 2025-02-13, 10:05:51
a03-p1-head-01	nvfw_GB200-P4978_0002_250205.1.0_dbg-signed.fwpkg	GB200-Switch	GB200-P4978_250205.1.0	available	N/A	16.2MiB 2025-02-05, 15:49:59
a03-p1-head-01	nvfw_GB200-P4978_0003_250121.1.2_custom_dbg-signed.fwpkg	GB200-Switch	GB200-P4978_250121.1.2_custom	available	N/A	1.64MiB 2025-01-21, 13:55:25

1. Use the firmware status command from the BCM device submenu to find the current firmware levels of the NVLink Switch

Example: Firmware Status Command from BCM

```
# Do for individual node
[a03-p1-head-01->device]% firmware status -n a05-p1-nvsw-09

# Do for all nodes
[a03-p1-head-01->device]% firmware status -t switch | grep -i nvsw

# Can also pull at the rack level if desired
[a03-p1-head-01->device]% firmware status -r <rack location> | grep -i nvsw
```

Example: Firmware Status Command Output

Device	State	Filename	Progress	Result	Component	Version
					Size	Date
→						
→						
a05-p1-nvsw-09		ASIC			35.2014.1698	current
→	N/A	N/A				
a05-p1-nvsw-09		BIOS			0ACTV_00.01.012	current
→	N/A	N/A				
a05-p1-nvsw-09		BMC			88.0002.0956	current
→	N/A	N/A				
a05-p1-nvsw-09		CPLD1			CPLD000370_REV0500	current
→	N/A	N/A				
a05-p1-nvsw-09		CPLD2			CPLD000377_REV0800	current
→	N/A	N/A				
a05-p1-nvsw-09		CPLD3			CPLD000373_REV0800	current
→	N/A	N/A				
a05-p1-nvsw-09		CPLD4			CPLD000390_REV0300	current
→	N/A	N/A				
a05-p1-nvsw-09		EROT			01.04.0018.0000_n04	current
→	N/A	N/A				
a05-p1-nvsw-09		EROT-ASIC1			01.04.0018.0000_n04	current
→	N/A	N/A				
a05-p1-nvsw-09		EROT-ASIC2			01.04.0018.0000_n04	current
→	N/A	N/A				
a05-p1-nvsw-09		EROT-BMC			01.04.0018.0000_n04	current
→	N/A	N/A				
a05-p1-nvsw-09		EROT-CPU			01.04.0018.0000_n04	current
→	N/A	N/A				
a05-p1-nvsw-09		EROT-FPGA			01.04.0018.0000_n04	current
→	N/A	N/A				
a05-p1-nvsw-09		FPGA			0.1A	current
→	N/A	N/A				
a05-p1-nvsw-09		SSD			CE00A400	current
→	N/A	N/A				
a05-p1-nvsw-09		transceiver			N/A	current
→	N/A	N/A				

2. Ensure that all NVLink Switch BMCs have their firmware management mode set to gb200sw.

```
#within CMSH
device
foreach -t switch (bmcsettings; get firmwaremanagemode)

#If not set
foreach -n S03-P1-NVSW-[01..09] (bmcsettings; set firmwaremanagemode
→gb200sw;commit)
```

3. To check against the versions in the firmware update file and ascertain if an update is needed, provide the file name in firmware flash -dry run command

```
#Single Switch
cmsh;device; firmware flash -n s03-p1-nvsw-04 nvfw_GB200-P4978-0007_
```

(continues on next page)

(continued from previous page)

```
→250121.1.2_custom_prod-signed.fwpkg --dry-run
```

#Multiple Switches

```
cmsh;device; firmware flash -n S03-P1-NVSW-[01-09] nvfw_GB200-P4978_0007_
→250121.1.2_custom_prod-signed.fwpkg --dry-run
```

4. If the changes look correct, then remove the `--dry-run` switch to apply the updates.
5. Update the tray level firmware first in this order:
 1. BMC+FPGA+ERoT (Switch BMC bundle).
 2. CPLD1 CPLD2 CPLD3 CPLD4 (Switch CPLD bundle).
 3. SBIOS+EROT (Switch BIOS bundle).
6. Use firmware status `-n <switch host name>` command to check update progress.
7. Once complete do an AC Cycle of the NVLink Switch to confirm the new firmware versions are active.

```
[a17-p1-bcm-01->device]% firmware status -n a18-p1-nvsw-09
```

Device	Filename	Component	Version	State	Progress	Result	Size	Date
a18-p1-nvsw-09	ASIC	35.2015.1686	current	N/A	N/A			
a18-p1-nvsw-09	BIOS	0ACTV_00.01.012	current	N/A	N/A			
a18-p1-nvsw-09	BMC	88.0002.0956	completed	N/A	success: activated	N/A		
a18-p1-nvsw-09	CPLD1	CPLD000370_REV0500	current	N/A	N/A			
a18-p1-nvsw-09	CPLD2	CPLD000377_REV0800	current	N/A	N/A			
a18-p1-nvsw-09	CPLD3	CPLD000373_REV0800	current	N/A	N/A			
a18-p1-nvsw-09	CPLD4	CPLD000390_REV0300	current	N/A	N/A			
a18-p1-nvsw-09	EROT	01.04.0018.0000_n04	completed	N/A	success: activated			
a18-p1-nvsw-09	EROT-ASIC1	01.04.0018.0000_n04	current	N/A	N/A			
a18-p1-nvsw-09	EROT-ASIC2	01.04.0018.0000_n04	current	N/A	N/A			
a18-p1-nvsw-09	EROT-BMC	01.04.0018.0000_n04	current	N/A	N/A			
a18-p1-nvsw-09	EROT-CPU	01.04.0018.0000_n04	current	N/A	N/A			
a18-p1-nvsw-09	EROT-FPGA	01.04.0018.0000_n04	current	N/A	N/A			
a18-p1-nvsw-09	FPGA	0.1A	current	N/A	N/A			
a18-p1-nvsw-09	SSD	CE00A400	current	N/A	N/A			
a18-p1-nvsw-09	transceiver	N/A	current	N/A	N/A			

1.2.3. Method 2 - Stand Alone nvfwupd Tool FW Update Process for NVLink Switch

Doing firmware updates with the nvfwupd tool is an alternative method to using the BCM firmware upgrade process. This method is highly manual.

1. To start do module load `cm-nvfwupd` (if the NVIDIA Mission Control enabled license is active), otherwise run the command from the location of the nvfwupd tool.
2. Assess NVLink Switch FW Levels from the nvfwupd tool.

```
nvfwupd -t ip=<switch IP> user=admin password=Juliet@1234
→servertime=gb200switch show_version
```

3. Compare the NVLink Switch versions found above with the versions in the update package.

```
nvfwupd -t ip=<switch IP> user=admin password=Juliet@1234
→servertime=gb200switch show_version -p <file to compare version to>
```

In this example all three NVLink Switch update files are passed to nvfwupdate to compare the versions of all upgradeable components.

```
root@T06-HEAD-01:~/nvfwup/release files v2.0.5/aarch64# ./nvfwupd -t ip=
→<NVLink Switch COMe0 IP> user=admin password=Juliet@1234
→servertime=gb200switch show_version -p ~/fw_0.9_releases/switch/nvfw_
→GB200-P4978_0004_250213.1.0_prod-signed.fwpkg ~/fw_0.9_releases/switch/
→nvfw_GB200-P4978_0006_250205.1.0_prod-signed.fwpkg ~/fw_0.9_releases/
→switch/nvfw_GB200-P4978_0007_250121.1.2_custom_prod-signed.fwpkg
```

```
System Model: N5400_LD
Part number: 920-9K36K-00MV-GS0
Serial number: MT250660041K
Packages: [ 'GB200-P4978_0004_250213.1.0', 'GB200-P4978_0006_250205.1.0',
→'GB200-P4978_0007_250121.1.2_custom' ]
Connection Status: Successful
```

Firmware Devices:

AP Name	Sys Version	Pkg Version	Up-To-Date

ASIC	35.2014.1652	N/A	No
BIOS	0ACTV_00.01.012	00.01.012	Yes
BMC	88.0002.0929	88.0002.0930	No
CPLD1	CPLD000370_REV0500	CPLD000370_REV0500	Yes
CPLD2	CPLD000377_REV0600	CPLD000377_REV0600	Yes
CPLD3	CPLD000373_REV0500	CPLD000373_REV0500	Yes
CPLD4	CPLD000390_REV0200	CPLD000390_REV0200	Yes
EROT	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
EROT-ASIC1	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
EROT-ASIC2	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
EROT-BMC	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
EROT-CPU	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
EROT-FPGA	01.04.0008.0000_n04	01.04.0008.0000_n04	Yes
FPGA	0.1A	0.1A	Yes
SSD	CE00A400	N/A	No
transceiver	N/A	N/A	No

```
ASIC 35.2014.1652 N/A No
BIOS 0ACTV_00.01.012 00.01.012 Yes
BMC 88.0002.0929 88.0002.0930 No
CPLD1 CPLD000370_REV0500 CPLD000370_REV0500 Yes
CPLD2 CPLD000377_REV0600 CPLD000377_REV0600 Yes
CPLD3 CPLD000373_REV0500 CPLD000373_REV0500 Yes
CPLD4 CPLD000390_REV0200 CPLD000390_REV0200 Yes
EROT 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-ASIC1 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-ASIC2 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-BMC 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-CPU 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-FPGA 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
FPGA 0.1A 0.1A Yes
SSD CE00A400 N/A No
transceiver N/A N/A No
```

```
→-----
Error Code: 0
```

4. Flash the NVLink Switch with the relevant package.

```
# Replace <switch IP> with the IP address of the switch
nvfwupd -t ip=<switch IP> user=admin password=Juliet@1234
→servertime=gb200switch update_fw -p /cm/local/apps/cmd/etc/htdocs/bios/
→firmware/gb200sw/nvfw_GB200-P4978_0000_241217.1.0_dbg-signed.fwpkg
```

5. Update the tray level firmware first in this order:
 1. BMC+FPGA+ERoT (Switch BMC bundle).
 2. CPLD1 CPLD2 CPLD3 CPLD4 (Switch CPLD bundle).
 3. SBIOS+EROT (Switch BIOS bundle).
6. After a BMC update, the switch will need an AC cycle.

Reference: NVLink Switch AUX Power Cycle using the nvfwupd tool- NVLink Switch NVUE power cycle

```
root@T06-HEAD-01:~/nvfwup/release files v2.0.5/aarch64# ./nvfwupd -t ip=10.78.
→195.1 user=admin password=Juliet@1234 servertype=gb200switch activate_fw -c
→NVUE_PWR_CYCLE
```

Power cycle task was created with ID 4

```
Status for Job Id 4:
{'detail': 'File delete successfully',
'http_status': 200,
'issue': [],
'percentage': '',
'state': 'running',
'status': 'File delete successfully',
'timeout': 5,
'type': '',
'warnings': []}
```

Note: The CPLD and SBIOS versions can be updated sequentially without a power cycle between them. The firmware update command will automatically trigger an AC cycle on the next reboot.

1. After reboot, check the firmware versions to ensure the update has completed.

Reference: NVLink Switch Successful BMC Update

```
root@T06-HEAD-01:~/nvfwup/release files v2.0.5/aarch64# ./nvfwupd -t ip=
→<NVLink Switch COMe0 IP> user=admin password=Juliet@1234
→servertype=gb200switch show_version -p ~/fw_0.9_releases/switch/nvfw_GB200-
→P4978_0004_250213.1.0_prod-signed.fwpkg ~/fw_0.9_releases/switch/nvfw_GB200-
→P4978_0006_250205.1.0_prod-signed.fwpkg ~/fw_0.9_releases/switch/nvfw_GB200-
→P4978_0007_250121.1.2_custom_prod-signed.fwpkg
```

```
System Model: N5400_LD
Part number: 920-9K36K-00MV-GS0
Serial number: MT250660041K
Packages: ['GB200-P4978_0004_250213.1.0', 'GB200-P4978_0006_250205.1.0',
→'GB200-P4978_0007_250121.1.2_custom']
Connection Status: Successful
```

```
Firmware Devices:
AP Name Sys Version Pkg Version Up-To-Date
-----
ASIC 35.2014.1652 N/A No
BIOS 0ACTV_00.01.012 00.01.012 Yes
BMC 88.0002.0930 88.0002.0930 Yes
CPLD1 CPLD000370_REV0500 CPLD000370_REV0500 Yes
```

(continues on next page)

(continued from previous page)

```

CPLD2 CPLD000377_REV0600 CPLD000377_REV0600 Yes
CPLD3 CPLD000373_REV0500 CPLD000373_REV0500 Yes
CPLD4 CPLD000390_REV0200 CPLD000390_REV0200 Yes
EROT 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-ASIC1 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-ASIC2 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-BMC 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-CPU 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
EROT-FPGA 01.04.0008.0000_n04 01.04.0008.0000_n04 Yes
FPGA 0.1A 0.1A Yes
SSD CE00A400 N/A No
transceiver N/A N/A No

```

 Error Code: 0

1.2.4. Method 3 - FW Updates within NVOS for NVLink Switch

In the event that the installed license does not support NVIDIA Mission Control feature, but updates need to be done anyway, it can be done using the NVOS itself. However it is firmly recommended that all other methods are attempted first, and use this as a last resort.

1. Assess NVLink Switch FW Levels from the NVOS

```
nv show platform firmware
```

Example: Login to NVLink Switch and Get Firmware/Software Version Info

```

#Firmware
admin@S04-P1-NVSW-01:~$ nv show platform firmware

Name Actual FW Part Number FW Source
-----
ASIC 35.2014.1652 920-9K36W-00MV-GS0_Ax default
BIOS 0ACTV_00.01.012 N/A N/A
BMC 88.0002.0929 692-13809-1404-000 N/A
CPLD1 CPLD000370_REV0500 0x0172 N/A
CPLD2 CPLD000377_REV0600 0x0179 N/A
CPLD3 CPLD000373_REV0500 0x0175 N/A
CPLD4 CPLD000390_REV0200 0x0186 N/A
EROT 01.04.0008.0000_n04 N/A N/A
EROT-ASIC1 01.04.0008.0000_n04 N/A N/A
EROT-ASIC2 01.04.0008.0000_n04 N/A N/A
EROT-BMC 01.04.0008.0000_n04 N/A N/A
EROT-CPU 01.04.0008.0000_n04 N/A N/A
EROT-FPGA 01.04.0008.0000_n04 N/A N/A
FPGA 0.1A N/A N/A
SSD CE00A400 Virtium VTPM24CEXI080-BM110006 N/A
transceiver N/A N/A N/A

```

Special Note on CPLD Upgrades (applicable to updating FW using NVOS only):

The CPLD archive is built into a ".fwpkg" package file type. To perform a CPLD upgrade on the NVLink Switch, unpack this file to obtain the ".vme" file required.

Download the NVIDIA "fwpkg-unpack" tool using PID 1090243.

Unpack the CPLD .fwpkg using the "fwpkg-unpack" tool:

```
./fwpkg-unpack -unpack nvfw_GB200-P4978_0007_250121.1.2_custom_prod-signed.fwpkg
```

A new CPLD file is extracted with a ".bin" file extension. Rename the file to have a ".vme" extension.

1. BMC Firmware update and Reboot (BMC + FPGA + ERoT)

```
nv action fetch platform firmware BMC 'scp://root:nvis1234!@192.168.255.
→254/var/www/html/NVLink Switch/images/0.9.03/nvfw_GB200-P4978_0004_
→250226.1.0_prod-signed.fwpkg'
```

```
nv action install platform firmware BMC files nvfw_GB200-P4978_0004_
→250226.1.0_prod-signed.fwpkg force
```

Note: System power cycle MUST be performed to force BMC to load the new FW version.

```
nv action power-cycle system force
```

2. CPLD Firmware update & Skip-Reboot (CPLD1 CPLD2 CPLD3 CPLD4)

```
nv action fetch platform firmware CPLD1 'scp://root:nvis1234!@192.168.255.
→254/var/www/html/nvswitch/images/0.9.03/CPLD_Prod_000370_REV0500_000377_
→REV0600_000373_REV0500_000390_REV0200_4717c08d_image.vme'
```

```
nv action install platform firmware CPLD1 files CPLD_Prod_000370_REV0500_
→000377_REV0600_000373_REV0500_000390_REV0200_4717c08d_image.vme force
→skip-reboot
```

3. BIOS Firmware Upgrade & Skip-Reboot (SBIOS + ERoT)

```
nv action fetch platform firmware BIOS 'scp://root:nvis1234!@192.168.255.
→254/var/www/html/nvswitch/images/0.9.03/nvfw_GB200-P4978_0006_250205.1.
→0_prod-signed.fwpkg'
```

```
nv action install platform firmware BIOS files nvfw_GB200-P4978_0006_
→250205.1.0_prod-signed.fwpkg force skip-reboot
```

1.3. NVLink Switch - Updating NVOS

For NVOS updates, outside of doing BCM ZTP automation, must be done on the NVLink Switch itself/NVOS.

1. Get NVOS Version. ssh to the admin user of the NVLink Switch and run the 'nv show system version' command.

```
#OS Software
admin@S04-P1-NVSW-01:~$ nv show system version
```

(continues on next page)

(continued from previous page)

```
operational
-----
kernel 5.10.0-30-2-amd64
build-date Sun Feb 9 18:12:03 UTC 2025
image nvos-25.02.1877
onie 2023.11-5.3.0012-115200
```

2. To install a new version of the NVOS, get the binary onto the host:

1. Use scp to get the binary to the switch and save the file in /host/nvos-images/
2. Or use the fetch command from NVOS to pull the .bin file

```
nv action fetch system image 'scp://root:nvis1234!@192.168.255.254/
→var/www/html/nvswitch/images/0.9.03/nvos-amd64-25.02.1884.bin'
```

3. Check system images that are present.

```
admin@S03-P1-NVSW-07:~$ nv show system image

operational
-----
current nvos-25.02.1877
next nvos-25.02.1877
partition1 nvos-25.02.1754
partition2 nvos-25.02.1877
```

4. Uninstall old images

1. Remove extra NVOS version image installed if present

```
nv action uninstall system image

admin@S03-P1-NVSW-07:~$ nv action uninstall system image

Action executing ...
Uninstalling image: nvos-25.02.1754
Action executing ...
Image nvos-25.02.1754 uninstalled successfully
Action succeeded
```

5. Install the new image. After the installation is completed, the switch will automatically reboot into the updated OS.

```
#nv action install system image files new-nvos-image.bin

admin@S03-P1-NVSW-07:~$ nv action install system image files nvos-amd64-
→25.02.1879.bin

The operation will install the image and initiate a reboot.

Type [y] to install the image and reboot.
Type [N] to abort.
```

(continues on next page)

(continued from previous page)

```
Do you want to continue? [y/N] y

Action executing ...
Installing image: nvos-amd64-25.02.1879.bin
Action executing ...
Performing reboot ...
Action executing ...
Disconnecting from NVOS, system is offline during reboot
Connection to s03-p1-nvsw-07 closed by remote host.
Connection to s03-p1-nvsw-07 closed.
```

6. When the switch OS comes back up after the reboot, check that the new OS version was applied using `nv show system image`.

```
admin@S03-P1-NVSW-07:~$ nv show system image

operational
-----
current nvos-25.02.1879
next nvos-25.02.1879
partition1 nvos-25.02.1877
partition2 nvos-25.02.1879
```

7. Check that the cluster apps are running on the switch that has been designated as the NMX-C master. This is typically NVSW-01.

```
admin@S04-P1-NVSW-01:~$ nv show cluster apps

Name ID Version Capabilities Components Version Status Reason Additional
→Information Summary
-----
-----
-----
nmx-controller nmx-c-nvos 0.9.0_2025-02-11_09-49 sm, gfm, fib, gw-api
→sm:2025.01.5, gfm:R570.120, fib-fe:0.9.0 ok CONTROL_PLANE_STATE_
→CONFIGURED
nmx-telemetry nmx-telemetry 0.9.5 nvl telemetry, gnmi aggregation, syslog
→aggregation nvl-telemetry:1.20.1, gnmi-aggregator:1.0.1, nmx-
→connector:1.0.1 ok
```

8. If this returns No data, and this is not the NMX-C master node, no further action is required. However, if the NVSwitch is the master the apps need to be configured within the NVOS:

```
# Start cluster apps
nv set cluster state enabled
nv config apply
nv config save
nv show cluster apps
```

1. If the NMX controller (NMX-C) is in the 'not ok' and says 'CONTROL PLANE_STATE_UNCONFIGURED', the `fm_config.cfg` file may need to be applied *per this section* where the `fm_config.cfg` file is generated.

```
admin@a18-p1-nvsw-01:~$ nv show cluster apps

Name ID Version Capabilities Components Version Status Reason Additional
→Information Summary
-----
-----
-----
nmx-controller nmx-c-nvos 0.9.0_2025-02-25_16-53 sm, gfm, fib, gw-api
→sm:2025.01.6, gfm:R570.124.02, fib-fe:0.9.0 not ok NMXC: OK CONTROL_
→PLANE_STATE_UNCONFIGURED
```

9. Re-run the litedaemon installation tool within BCM in order for the switch to show UP.

Note: Sometimes after a new NVOS installation, the default factory password gets reset to admin. Login with admin/admin, set the password to [Juliet@1234](#) and then try again.

Example: NVOS Default State, Password Reset

NVOS switch

admin@s03-p1-nvsw-04's password:

You are required to change your password immediately (administrator enforced).

Last login: Fri Mar 21 08:58:02 UTC 2025 from 10.78.192.25 on pts/0

Last failed login: Fri Mar 21 10:02:38 UTC 2025 from 10.78.192.25 on ssh:notty

There was 1 failed login attempt since the last successful login.

WARNING: Your password has expired.

You must change your password now!

New password:

Retype new password:

applied [rev_id: 1]

Number of total successful connections since last 1 days: 3

Your password has been changed since last login

Note

A pause is expected after the new password change.

Chapter 2. Power Shelf Firmware Update Process

The following instructions are for the power shelves on DGX GB200.

Table 1: Power Shelf Components

No	Series	Description	Part Number (PN)
1	Delta PR1	19" 1RU PSU 5.5kW-DGX	ECD17020036
2	Delta PR1	19" 1RU PMC-DGX	ECD70000011
3	Delta PR3	19" 1RU PSU 5.5kW-DGX	ECD17020044
4	Delta PR3	19" 1RU PMC-DGX	ECD70000018

PMC FW Update for Delta PR1

1. Get the Power Shelf part number:

```
curl -k -u $USER:$PASS -X GET https://<BMC_IP>/redfish/v1/Managers/PMC_0
```

Use the Part Number in the table above to make sure you have a Delta PR1 PowerShelf.

2. Get PMC FW version for the Primary and Backup partitions.

```
curl -sk -u $USER:$PASS -X GET https://${PMC_IP}/redfish/v1/Managers/PMC_0 | grep Firm | egrep 'Prima|Back'
```

3. Set the FlashSelector to Backup

```
curl -sk -u $USER:$PASS -X PATCH https://${PMC_IP}/redfish/v1/Managers/PMC_0 --data '{"Oem" : {"deltaenergysystems": {"FlashSelector": "Backup"}}}' -H "content-type: application/json"
```

4. Set the ApplyTime setting based on your preference - Immediate (Recommended) or OnReset

```
curl -k -u $USER:$PASS https://${PMC_IP}/redfish/v1/UpdateService -X PATCH -d '{"HttpPushUriOptions": {"HttpPushUriApplyTime": {"ApplyTime": "Immediate"}}}'
```

5. Update FW for the Backup partition

```
curl -k -u $USER:$PASS -X POST -H "Content-Type: application/octet-stream" --data-binary "@${PMC_FW_FILE}" https://${PMC_IP}/redfish/v1/UpdateService/update
```

6. Set the FlashSelector to Primary

```
curl -sk -u $USER:$PASS -X PATCH https://${PMC_IP}/redfish/v1/Managers/PMC_0 --data '{"Oem" : {"deltaenergysystems": {"FlashSelector": "Primary"}}}' -H "content-type: application/json"
```

7. Update FW for the Primary partition

```
curl -k -u $USER:$PASS -X POST -H "Content-Type: application/octet-stream" --data-binary "@${PMC_FW_FILE}" https://${PMC_IP}/redfish/v1/UpdateService/update
```

8. Confirm FW version for both primary and backup partitions

```
curl -sk -u $USER:$PASS -X GET https://${PMC_IP}/redfish/v1/Managers/PMC_0 | grep Firm | egrep 'Prima|Back'
```

PMU FW Update for Delta PR3

1. Get the Power Shelf part number:

```
curl -k -u $USER:$PASS -X GET https://<BMC_IP>/redfish/v1/Managers/PMC_0
```

Use the Part Number in the table above to make sure you have a Delta PR3 PowerShelf.

2. Get PMC FW version for the Primary and Backup partitions.

```
curl -sk -u $USER:$PASS -X GET https://${PMC_IP}/redfish/v1/Managers/PMC_0 | grep Firm | egrep 'Prima|Back'
```

3. Set the ApplyTime setting based on your preference - Immediate (Recommended) or OnReset

```
curl -k -u $USER:$PASS https://${PMC_IP}/redfish/v1/UpdateService -X PATCH -d '{"HttpPushUriOptions": {"HttpPushUriApplyTime": {"ApplyTime": "Immediate"}}}'
```

4. Update FW for the Backup partition - Use the Backup PMC FWpkg file. Ex: "nvidia-pmc-3.*-RELEASE.FOR.BACKUP.UPDATE.CEC-CFG.fwpkg"

```
curl -k -u $USER:$PASS -X POST -H "Content-Type: application/octet-stream" --data-binary "@${PMC_FW_BACKUP_FILE}" https://${PMC_IP}/redfish/v1/UpdateService/update
```

5. Update FW for the Primary partition - Use the Primary PMC FWpkg file. Ex: "nvidia-pmc-3.*-RELEASE.FOR.UPDATE.CEC-CFG.fwpkg"

```
curl -k -u $USER:$PASS -X POST -H "Content-Type: application/octet-stream" --data-binary "@${PMC_FW_PRIMARY_FILE}" https://${PMC_IP}/redfish/v1/UpdateService/update
```

6. Confirm FW version for both primary and backup partitions

```
curl -sk -u $USER:$PASS -X GET https://{PMC_IP}/redfish/v1/Managers/PMC_0 | grep Firm | egrep 'Prima|Back'
```

PSU FW Update

1. Confirm the current PSU firmware version:

```
curl -k -u $USER:$PASS -X GET https://<BMC_IP>/redfish/v1/Managers/PMC_0 | grep Version
```

2. Check the PSU part number to ensure it matches the table above:

```
curl -k -u $USER:$PASS -X GET https://<BMC_IP>/redfish/v1/Managers/PMC_0 | grep PartNumber
```

3. Flash the PSU firmware using the same command from PMC step 5 with <FIRMWARE_FILE> replaced with PSU firmware binary
4. Run step 1 again to ensure the PSU firmware updated successfully

Note

PMC FW update timing can be customized by the ApplyTime property:

- **Immediate (Default)** - Recommended. Flashes the PMC immediately.
- **OnReset** - Update to new firmware version on the next reboot.

GET ApplyTime property:

```
curl -k -u $USER:$PASS -X GET https://<BMC_IP>/redfish/v1/UpdateService
```

SET ApplyTime property to Immediate:

```
curl -k -u $USER:$PASS -X PATCH 'https://<BMC_IP>/redfish/v1/UpdateService' \
-d '{"HttpPushUriOptions": {"HttpPushUriApplyTime": {"ApplyTime": "Immediate"}}}' \
-H 'Content-Type: application/json'
```

Warning

A PSU firmware update will temporarily power off the PSU, so we recommend that the rack is idle during the PSU update process.

Chapter 3. Bluefield and CX7 Firmware Update Process

Prior to installation, copy the binary to the compute tray host or use a shared directory. The binary naming format should look like:

```
fw-ConnectX7-rel-28_42_1270-900-24768-0002_Ax-UEFI-14.35.15-FlexBoot-3.7.500.  
→signed.bin
```

The general steps to install NVIDIA Networking firmware are as follows:

1. Start the MST service

```
mst start
```

2. Query the devices to find the /dev/mst paths of the devices

```
mst status -v
```

3. Read the current version of firmware on a given device

```
flint -d /dev/mst/mt4129_pciconf0 q full
```

4. Flash the firmware on the device

- a. Change to the directory where the binary is stored
- b. Run the flash command:

```
flint -d /dev/mst/mt4129_pciconf0 -i fw-ConnectX7-rel-28_42_1270-900-  
→24768-0002_Ax-UEFI-14.35.15-FlexBoot-3.7.500.signed.bin b
```

- c. Repeat this for all 4 CX7 devices

5. Reset the CX7 and reboot the host

```
mlxfwreset -d mlx5_0 reset
```

PCI devices:

DEVICE_TYPE	MST	PCI	RDMA	NET	NUMA
ConnectX7(rev:0)	/dev/mst/mt4129_pciconf0	0000:03:00.0	mlx5_0	net-ibp3s0	0
ConnectX7(rev:0)	/dev/mst/mt4129_pciconf1	0002:03:00.0	mlx5_1	net- →ibP2p3s0	0

(continues on next page)

(continued from previous page)

```
ConnectX7(rev:0) /dev/mst/mt4129_pciconf2 0010:03:00.0 mlx5_4 net-
→ibP16p3s0 1
ConnectX7(rev:0) /dev/mst/mt4129_pciconf3 0012:03:00.0 mlx5_5 net-
→ibP18p3s0 1
```

6. For Bluefield 3, the process is the same with the exception of the device being /dev/mst/mt41692.

3.1. Combined CX-7 and Bluefield Update

Run the update script across all GB200 nodes:

```
pdsh -g category=dgx-gb200 '/home/nvis/(dir where the firmware update is)/
→nicupdate.sh > /home/nvis/(dir where the firmware update is)/$(hostname)_fw_
→upgrade_$(date +%Y%m%d-%H%M%S').log'
```

Reference Script: NIC Updates (Both BF3 and CX-7) - nicupdate.sh

```
# CX-7 Update
mst start

flint -d /dev/mst/mt4129_pciconf0 q full
flint -d /dev/mst/mt4129_pciconf1 q full
flint -d /dev/mst/mt4129_pciconf2 q full
flint -d /dev/mst/mt4129_pciconf3 q full

# BF3 Update
flint -d /dev/mst/mt41692_pciconf0 q full
flint -d /dev/mst/mt41692_pciconf1 q full

basedir=/cm/shared/nicfw
bf3file=fw-BlueField-3-rel-32_44_1600-900-9D3B6-00CN-P_Ax-NVME-20.4.1-UEFI-21.
→4.13-UEFI-22.4.14-UEFI-14.37.14-FlexBoot-3.7.500.signed.bin
cx7file=fw-ConnectX7-rel-28_44_2524-900-24768-0002_Ax-UEFI-14.37.14-FlexBoot-
→3.7.500.signed.reduced.bin

yes | flint -d /dev/mst/mt4129_pciconf0 -i $basedir/$cx7file b
yes | flint -d /dev/mst/mt4129_pciconf1 -i $basedir/$cx7file b
yes | flint -d /dev/mst/mt4129_pciconf2 -i $basedir/$cx7file b
yes | flint -d /dev/mst/mt4129_pciconf3 -i $basedir/$cx7file b

yes | flint -d /dev/mst/mt41692_pciconf0 -i $basedir/$bf3file b
yes | flint -d /dev/mst/mt41692_pciconf1 -i $basedir/$bf3file b
```

Chapter 4. Software Update

1. Clone the category in BCM

```
cmsh
category
clone <dgx-gb200> <new-dgx-gb200>
commit
```

2. Clone the OS image

```
cmsh
softwareimage
clone <dgx-image> <new-dgx-image>
commit
```

Set the new Category to the new image

```
cmsh
category
set <new-dgx-gb200> softwareimage <new-dgx-image>
commit
```

3. Enter the Image to make changes

```
cm-chroot /cm/images/new-dgx-image/
```

4. Create DOCA Repo based on Architecture

X86:

```
dd status=none of=/etc/apt/sources.list.d/doca.sources << EOF
Types: deb
URIs: https://linux.mellanox.com/public/repo/doca/baseos8-latest/ubuntu24.
      ↪04/x86_64/
Suites: /
Signed-By: /usr/share/keyrings/GPG-KEY-Mellanox.gpg
EOF
```

arm64:

```
dd status=none of=/etc/apt/sources.list.d/doca.sources << EOF
Types: deb
URIs: https://linux.mellanox.com/public/repo/doca/baseos8-latest/ubuntu24.
```

(continues on next page)

(continued from previous page)

```
→04/arm64-sbsa/  
Suites: /  
Signed-By: /usr/share/keyrings/GPG-KEY-Mellanox.gpg  
EOF
```

5. Install the latest DGX OS packages.

For a detailed DGX OS update guide, please refer to <https://docs.nvidia.com/dgx/dgx-os-7-user-guide/upgrading-the-os.html#performing-package-upgrades-using-the-cli>

```
1. Update the internal database with the list of available packages and  
→their versions.  
apt update  
  
2. Review the packages that will be upgraded.  
apt full-upgrade -s  
  
3. Upgrade to the latest version.  
apt full-upgrade
```

Verify installations:

```
# Check DOCA packages  
sudo dpkg -l | grep <Expected DOCA Ver>  
  
# Check driver package  
sudo dpkg -l | grep <Expected Driver ver>
```

6. Save changes into the image

```
exit
```

7. Set compute node to DGX Category

```
cmsh  
device  
foreach -n dgx-nodes[XX-XX] (set category <new-dgx-gb200>)  
commit
```

8. Reboot compute nodes

```
reboot -c <new-dgx-gb200>
```

9. Verify all components have been upgraded

Chapter 5. Operational Security Requirements

5.1. Factory Reset after Debug Token Usage

After using one (or more) Debug Token on the compute tray, the operator **must** remove the Debug Token and factory reset the non-volatile storage of the BMC, HMC, and CPU. The following RedFish APIs provide the factory reset functionality:

Reset the HMC R/W filesystem:

```
curl -k -u $USER:$PASS -X POST \
https://${TARGET_HOSTNAME}/redfish/v1/Managers/HGX_BMC_0/Actions/Manager.
  ↳ ResetToDefaults \
-d '{"ResetToDefaultsType": "ResetAll"}'
```

Erase the HMC eMMC:

```
curl -k -u $USER:$PASS -X POST \
https://${TARGET_HOSTNAME}/redfish/v1/Managers/HGX_BMC_0/Actions/Oem/eMMC.
  ↳ SecureErase
```

Erase the Grace R/W SPI flashes (both modules):

```
curl -k -u $USER:$PASS -X POST \
https://${TARGET_HOSTNAME}/redfish/v1/Chassis/HGX_ProcessorModule_0/Actions/
  ↳ Oem/NvidiaProcessor.VariableSpiErase

curl -k -u $USER:$PASS -X POST \
https://${TARGET_HOSTNAME}/redfish/v1/Chassis/HGX_ProcessorModule_1/Actions/
  ↳ Oem/NvidiaProcessor.VariableSpiErase
```

Reset the BCM R/W filesystem:

```
curl -k -u $USER:$PASS -X POST \
https://${TARGET_HOSTNAME}/redfish/v1/Managers/BMC_0/Actions/Manager.
  ↳ ResetToDefaults \
-d '{"ResetToDefaultsType": "ResetAll"}'
```

Erase the BMC eMMC:

```
curl -k -u $USER:$PASS -X POST \  
https://${TARGET_HOSTNAME}/redfish/v1/Managers/BMC_0/Actions/Oem/eMMC.  
→SecureErase
```

Chapter 6. Additional Information

6.1. Compute Tray BMC Username

Check that the BMC has the username `admin` installed. From the compute tray host OS, run:

```
ipmitool user list 1 | grep admin
```

From the BCM console to check the username:

```
cmsh; device use <device-name> ; bmcsettings ; get username
```

If the username in use is not `admin`, use the following commands from the compute tray host OS to create it:

```
ipmitool user set name 3 admin
ipmitool user set password 3 GetNVIDIA2025!
ipmitool channel setaccess 1 3 link=on ipmi=on callin=on privilege=4
ipmitool user enable 3
sleep 5
ipmitool user list 1 | grep admin
```

Alternatively, you can check currently enabled users through the BMC Web GUI by going to **Security and access >> User Management**.

Overview

Logs

Hardware status

Operations

Settings

Security and access

Sessions

LDAP

User management

Policies

Certificates

User management

Account policy settings + Add user

☐	Username	Privilege	Status	
☐	admin	Administrator	Enabled	
☐	root	Administrator	Disabled	

View privilege role descriptions

You can also add the **admin** account with Administrator privilege through this interface by selecting **Add User**.

Add user



Account status

☒ Enabled☐ Disabled

Username

Cannot start with a number
No special characters except
underscore

admin

User password

Password must be between 13 –
64 characters



Confirm user password



Privilege

Administrator



Cancel

Add user

Copyright

©2022-2025, NVIDIA Corporation