

Device firmware upgrade

Via CLI/CURL

How to use CNaas NMS for firmware upgrades? Well, this guide will try to describe how to do it step by step.

Any hostnames MUST be replaced with your own hostnames or IP addresses.

First, check if the firmware is already downloaded. If it's already present, there is no need to download it again.

CLI:

```
CNaas NMS (norpan-nms)# show files
EOS-4.24.2.3F.swi
```

CURL:

```
: 18:00 root@norpan-nms: ~ # curl -s -k -H "Authorization: Bearer ${JWT_AUTH_TOKEN}" https://norpan-nms.cnaas.io/api/v1.0/firmware | jq .
{
  "status": "success",
  "data": {
    "files": [
      "EOS-4.24.2.3F.swi"
    ]
  }
}
```

If the firmware is not present in the previous step, we must download it.

CLI:

```
CNaas NMS (norpan-nms)# download url https://test.example.com/EOS-4.24.2.3F.swi sha1
18ed4efc42933a66c62a3ba90aeb31d6e2ad208c verify_tls true
```

CURL:

```
: 18:00 root@norpan-nms: ~ # curl -s -k -H "Authorization: Bearer ${JWT_AUTH_TOKEN}" https://norpan-nms.cnaas.io/api/v1.0/firmware -X POST -H "Content-Type: application/json" -d '{"url": "https://test.example.com/EOS-4.24.2.3F.swi", "sha1": "18ed4efc42933a66c62a3ba90aeb31d6e2ad208c", "verify_tls": false}'
```

In the commands above, we have three different arguments:

- url: The URL from where we can download the firmware image.
- sha1: The SHA1 checksum of the firmware, we use this to verify that we really downloaded it.
- verify_tls: If we should ignore invalid SSL certificates or not.

Upgrade the devices

The last step is to upgrade the devices.

CLI:

```
CNaas NMS (norpan-nms)# firmware_upgrade hostname a1 filename EOS-4.24.2.3F.swi url http://norpan-nms.cnaas.io/firmware/ pre_flight true download true reboot true activate true post_flight true post_waittime 600
```

CURL:

```
curl -k -s -H "Content-Type: application/json" -H "Authorization: Bearer ${JWT_AUTH_TOKEN}" -X POST https://norpan-nms.cnaas.io/api/v1.0/firmware/upgrade -d '{"hostname": "a1", "filename": "EOS-4.24.2.3F.swi", "url": "http://norpan-nms.cnaas.io/firmware/", "pre-flight": true, "download": true, "activate": true, "reboot": true, "post_flight": true, "post_waittime": 600}'
```

As can be seen above, we have a few arguments here as well:

- hostname: The hostname of the device to upgrade. This can be replaced with group if you want to upgrade one or more devices.
- filename: The filename of the firmware to install.
- url: The URL from where the DEVICE can download the firmware, should be the NMS server we downloaded the firmware to in a previous step.
- pre_flight: Do some basic checks before upgrading. Check free diskspace etc.
- download: If we should download the firmware or not.
- activate: If we should install the firmware or not.
- reboot: If the device(s) should be rebooted or not.

- `post_flight`: Update the OS version etc in the NMS database after the firmware is installed.
- `post_waittime`: How long to wait (in seconds) before trying to perform the `post_flight` step. This is needed since it can take up to 10 minutes for a device to reboot.

WebUI

Demo: https://play.sunet.se/media/CNaaS-NMS+Firmware+Upgrade+via+WebUI/0_jf424rq1

Firmware can only be downloaded to the NMS instance via the SUNET CNaaS Firmware repository when using the WebUI.

When the correct firmware is available in the firmware repository you can use the WebUI to download it to the local NMS instance by clicking the "Firmware" link in the navigation menu in the WebUI. Find the firmware you want to use in the list and click "Copy to NMS" if it's not already downloaded to the local NMS instance.

You can then proceed to the "Groups" page in the WebUI and find the group of devices you want to upgrade. Click the "Firmware upgrade..." link for the corresponding group and it will take you to a separate page for the firmware upgrade.

Verify the currently running versions of devices in the group and make sure they are compatible for upgrade. The eos.arista.com website has a "MLAG compatibility" check you can use if upgrading devices that are part of an MLAG group for example.

When you are ready to start please include a service ticket/change reference ID at the start of the page so other people can find more information about the upgrade job later.

Select the firmware you want to use for the device group in the dropdown list and then click "Start activate firmware". A job will be scheduled to download and activate the firmware to the devices. The progress bar only updates when a device has completed the download, you can't see the actual speed of the download on each individual device.

If the firmware download/activate was successful, or if it was already activated, you can proceed to the final step which is to reboot the devices. For Arista the downtime is usually around 6-10 minutes(?) so keep that in mind. It might also be a good idea to start a `downtimereporter.py` script to measure the downtime. You can either start the reboots immediately by not filling out a date/time, or schedule it to occur at a later time by using the field. The abort button can't really do anything to stop devices that are already in the process of restarting, but it can be used if you want to stop the scheduled job.

If you leave the firmware upgrade page you will not be able to browse back to see progress of running/scheduled jobs, but you can find those under "Jobs" in the navigation menu. You also can't abort scheduled jobs if you leave the page, instead you will have to use CURL if you want to abort a job in that case.