NVIDIA NVOS User Manual for InfiniBand Switches v25.03.1010

Table of Contents

Welcome to NVOS Documentation

NVIDIA NVOS operating system enables the management and configuration of NVIDIA's switch system platforms.
NVOS provides a suite of management options, incorporates a CLI and OpenAPI, which enables administrators to easily configure and manage the system.

These pages provide information about the scope, organization, and command-line interface of NVOS as well as configuration examples.

Software Download

To download the latest software, log in to the following website: enterprise-support.nvidia.com/s/

For common questions about the Enterprise Account please see the following webpage: nvid.nvidia.com/NvidiaUtilities/#/needHelp

Technical Support

- Customers who purchased NVIDIA products directly from NVIDIA can contact support via the Enterprise Support page: https://www.nvidia.com/en-us/support/enterprise
- Customers who purchased NVIDIA M-1 Global Support Services, please see your contract for details regarding technical support.
- Customers who purchased NVIDIA products through an NVIDIA-approved reseller should first seek assistance through their reseller.

Document Revision History

A list of the changes made to the User Manual is provided in Document Revision History section.

1 Overview

1.1 Intended Audience

These pages are intended for network administrators who are responsible for configuring and managing NVOS platforms.

1.2 Related Documentation

The following table lists the documents referenced in this user manual.

Document Name	Description
Q32xx and Q34xx XDR 800Gb/s InfiniBand Switch Systems User Manual	This manual describes the installation and basic use of the NVIDIA XDR InfiniBand switch systems based on the NVIDIA Quantum™-3 switch ASIC
NVIDIA NVOS Release Notes	Provides information about the supported platforms, changes and new features, software known issues, and bug fixes. See the Enterprise Support Portal for more information.

1.3 Terminology

Term	Description
AAA	Authentication, Authorization, and Accounting. A security framework for controlling and tracking user access within a computer network. Authentication verifies user credentials; Authorization grants or denies privileges; Accounting tracks user activities and resource consumption.
ACL	Access Control List. A set of filtering rules applied to interfaces or ports that control network traffic based on source/destination IP, port number, or protocol type.
ARP	Address Resolution Protocol. Maps IP addresses to physical MAC addresses for communication within a local area network (LAN).
ASIC	Application-Specific Integrated Circuit. A custom-designed chip optimized for specific functions such as packet forwarding in network switches.
BIOS	Basic Input/Output System. Firmware that initializes and tests hardware components during system boot and loads the operating system.
BMC	Baseboard Management Controller. A service processor that enables out-of-band hardware monitoring and management (temperature, fan speed, power).
CLI	Command-Line Interface. A text-based interface used to configure, manage, and monitor the switch through command input.

Core Dump	A file containing memory and process state at the time of a crash, used for debugging.
Crashdump	Diagnostic data collected automatically after a system or process failure.
CPLD	Complex Programmable Logic Device. A programmable logic device used for control logic, often handling board-level management and power sequencing.
DHCP	Dynamic Host Configuration Protocol. Automatically assigns IP addresses and network configuration parameters to devices on a network.
DNS	Domain Name System. Translates human-readable domain names into corresponding IP addresses for network communication.
ERoT	External Root of Trust. A secure cryptographic hardware component that verifies system integrity and authenticates firmware during boot.
EEPROM	Electrically Erasable Programmable Read-Only Memory. Non-volatile memory used to store firmware or configuration data that must persist across power cycles.
Ethernet	A family of networking technologies used for LAN communication, defining physical and data link layer protocols.
FPGA	Field Programmable Gate Array. A reconfigurable integrated circuit that can be programmed post-manufacturing to implement custom logic or accelerate specific functions.
Fabric	The interconnected topology of switches and links forming a unified network for data transfer, commonly used in high-performance environments.
FRU	Field Replaceable Unit. A hardware component that can be replaced without special tools, typically including fans, power supplies, and management modules.
FTP / TFTP / SFTP	File Transfer Protocols. Protocols used to transfer files between systems. FTP uses TCP; TFTP is simplified and connectionless; SFTP provides secure encrypted transfer over SSH.
gNMI	gRPC Network Management Interface. A streaming-based protocol for network configuration and telemetry using gRPC, supporting both configuration and real-time monitoring.
GUI	Graphical User Interface. A visual interface allowing users to interact with and configure the system via icons, menus, and windows rather than command lines.
Host	A device connected to a network that can send, receive, and process data with other network devices.
ICMP	Internet Control Message Protocol. Used to send control messages and error notifications between network devices, commonly used for ping tests.
I^2C	Inter-Integrated Circuit. A low-speed serial communication bus used internally on circuit boards for connecting components such as sensors, EEPROMs, or CPLDs.
LACP	Link Aggregation Control Protocol. A protocol for combining multiple physical network links into a single logical interface for redundancy and performance.
LAG	Link Aggregation Group. A logical grouping of multiple physical links to increase bandwidth and provide redundancy.
Loopback Interface	A virtual interface used for testing, routing, and management reachability.
MAC	Media Access Control Address. A unique hardware identifier assigned to each network interface for communication on the physical layer.
MTU	Maximum Transmission Unit. The largest payload size (in bytes) that can be sent in a single packet without fragmentation.

NTP	Network Time Protocol. Synchronizes system clocks of devices across a network to maintain accurate timekeeping.
NVRAM	Non-Volatile Random Access Memory. Memory that retains data even when power is turned off, often used for configuration storage.
NVLink	A high-speed interconnect developed by NVIDIA that allows direct communication between GPUs or between GPU and CPU.
NVOS	NVIDIA Operating System. Provides system and network management functionality for NVLink switches.
Network Adapter	A hardware component that enables communication between a computer or device and a network.
Overlay Network	A virtual network built on top of a physical underlay to abstract or isolate traffic, e.g., VXLAN-based networks.
PCIe	Peripheral Component Interconnect Express. A high-speed interface standard for connecting hardware components like NICs and GPUs.
QoS	Quality of Service. A set of mechanisms that prioritize or limit network traffic based on policies to ensure performance for critical applications.
REST API	Representational State Transfer API. A web-based interface that allows external systems to configure and monitor devices using HTTP/HTTPS requests.
Routing Table	A data table stored in a switch or router that lists paths to particular network destinations.
SA	Subnet Agent. A process running on each node that communicates with the Subnet Manager (SM) to maintain fabric topology and routing data.
SCP	Secure Copy Protocol. A secure file transfer method that uses SSH to copy files between local and remote hosts.
SM	Subnet Manager. The central process that initializes and maintains the InfiniBand or NVLink subnet, assigns local IDs (LIDs), and manages routing.
SNMP	Simple Network Management Protocol. Used for monitoring and managing network devices and collecting operational data.
SNTP	Simple Network Time Protocol. A simplified version of NTP for basic clock synchronization.
SPI	Serial Peripheral Interface. A synchronous serial communication interface used for short-distance communication between microcontrollers and peripherals.
SSH	Secure Shell. A cryptographic network protocol enabling secure remote login and command execution between devices.
syslog	A standard protocol for sending system log or event messages to a centralized log server for monitoring and analysis.
TACACS+	Terminal Access Controller Access-Control System Plus. A protocol that provides centralized authentication, authorization, and accounting for network device access.
TPM	Trusted Platform Module. A hardware chip that provides hardware-based security, cryptographic key storage, and secure boot validation.
Underlay Network	The physical network that provides IP connectivity for overlay networks or tunnels.
VRF	Virtual Routing and Forwarding. Enables multiple routing tables to coexist on the same switch, allowing network segmentation and traffic isolation.
VLAN	Virtual Local Area Network. A logical grouping of devices that communicate as if on the same LAN, even if physically separated.

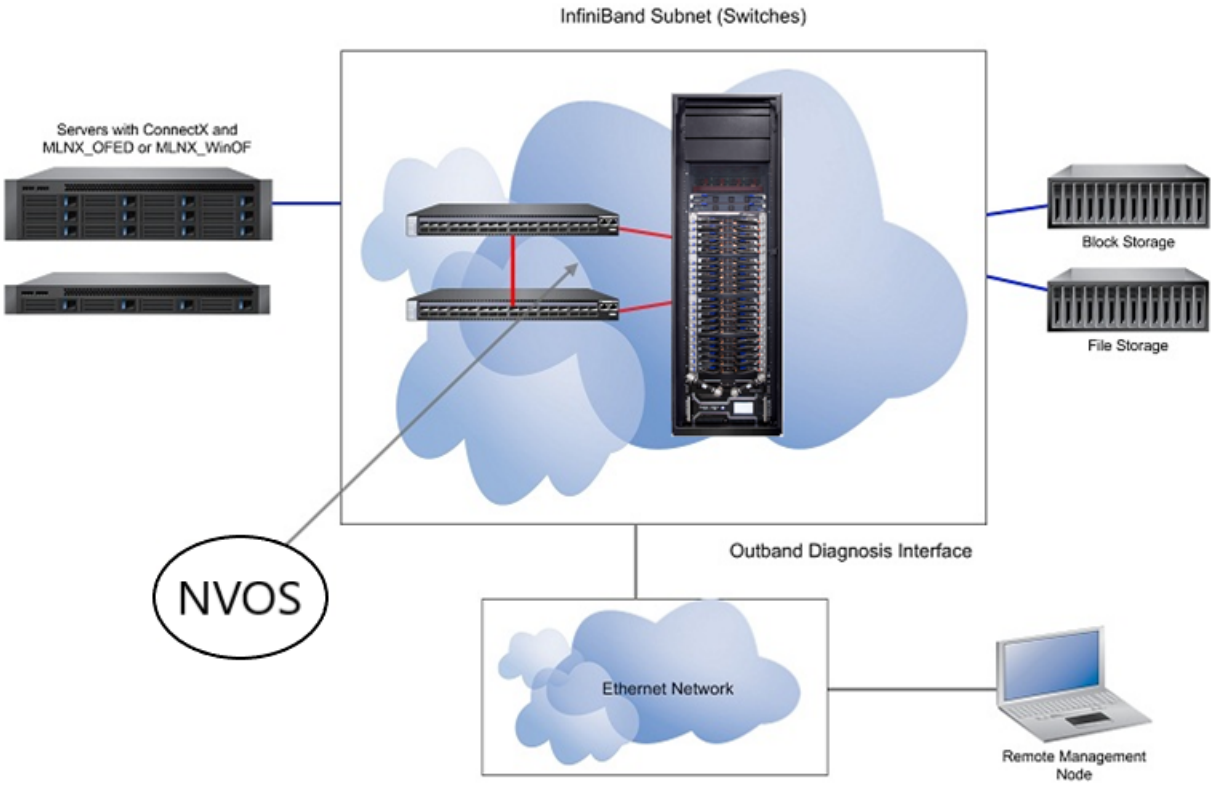
Watchdog Timer	A hardware or software timer that resets the system if it becomes unresponsive.
ZTP	Zero-Touch Provisioning. Automates initial device configuration by allowing the switch to fetch and apply configuration files and firmware on first boot.

1.4 System Features

Feature	Detail
Software management	• Software updates • Firmware updates
Logging	• System log • Tech-support
Management interface	• IP • DHCP • Hostname
Chassis management	• Monitoring environmental controls
Network management interfaces	• OpenAPI
Security	• Password Hardening
Cables & transceivers	• Transceiver info

1.5 InfiniBand Features

Feature	Detail
Subnet manager	• OpenSM
IB port management	• IB port interfaces
IB Fabric	• IB devices


InfiniBand Subnet (Switches)
Servers with ConnectX and MLNX_OFED or MLNX_WinOF
Block Storage
File Storage
NVOS
Outband Diagnosis Interface
Ethernet Network
Remote Management Node

2 Quick Start Guide

This quick start guide provides an end-to-end setup process for installing and running NVOS.

> ⚠ Before running NVUE commands, please refer to the NVIDIA User Experience (NVUE) section.

> ⚠ NVOS supports configurations through NVUE only. Running Linux commands or directly modifying underlying Linux files might result in undefined behavior.

2.1 Prerequisites

This guide requires an intermediate-level Linux knowledge. An understanding of text editing, Unix file permissions, and process monitoring is necessary. A variety of text editors are pre-installed, including `vi` and `nano`.

Access to a Linux or UNIX shell in needed. For Windows users, employing a Linux environment like Cygwin as your command line tool is recommended for interacting with NVOS.

> ⚠ If moving from MLNX-OS to NVOS operating system, please refer to the following appendix: Moving from MLNX-OS to NVOS.

2.2 Get Started

NVOS provides two methods for initial provisioning:

- Zero-Touch Provisioning (ZTP)
- Manual Provisioning

2.3 Manual Provisioning

When starting NVOS for the first time, the management port send a DHCP request. To determine the IP address of the switch, you can cross reference the MAC address of the switch with your DHCP server. The MAC address is typically located on the side of the switch or on the box in which the unit ships.

2.3.1 Login Credentials

The default installation includes two accounts:

- The user account (admin) has **sudo** privileges. The admin account uses the default password **admin**.
- The user account (monitor) has **read only** privileges. The monitor account uses the default password **monitor**.

Upon first login, it will be required to change the default passwords for the admin and monitor accounts. The new password must comply with the default password hardening rules (see Password Hardening section).

The new configured password is treated like any other configuration. If the configuration is not saved, the user will need to reconfigure it upon the system's next boot.

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




WARNING: Your password has expired.
You must change your password now!
New password:
Retype new password:
```

In this quick start guide, use the *admin* account to configure NVOS.

All accounts except root can use remote SSH login. Note that the default SSH auto-logout is 15 minutes.

NVOS supports multiple services to authenticate users and authorize switch tasks, both local and remote authentication/authorization methods.
For more information, please refer to Authentication Authorization and Accounting section.

2.3.2 Serial Console Management

It is recommended to perform management and configuration over the network, either in-band or out-of-band. A serial console is fully supported.

Typically, switches ship from the manufacturer with a mating DB9 serial cable. Switches with ONIE are always set to a 115200 baud rate.

Recommended Serial Connection Settings

Parameter	Setting
Baud Rate	115200
Data bits	8
Stop bits	1
Parity	None
Flow Control	None

2.3.3 Wired Ethernet Management

An NVOS switch always provides two dedicated Ethernet management port called eth0 and eth1. This interface is specifically for out-of-band management use. The management interface uses DHCP for addressing by default.

2.3.4 Set Static IP Address

To set a static IP address, run the following (ipv6 supported as well):

```
admin@nvos:~$ nv set interface eth0 ipv<4/6> address 192.0.2.42/24
admin@nvos:~$ nv set interface eth0 ipv<4/6> gateway 192.0.2.1
admin@nvos:~$ nv config apply
```

Configuring static IP address will trigger unsolicited announcement messages to the gateway in order to reveal the devices' MAC address.

2.3.5 Configure the Hostname

The hostname identifies the switch; as such, make sure the hostname is configured in a unique and descriptive way. For more information, see Hostname section.

2.3.6 Configure the System Clock

NVOS relies on the system clock for displaying the time and timestamping messages. User can control the system time zone and clock settings.

2.3.6.1 Manual Settings

2.3.6.1.1 Time Zone

Default time zone is set to Coordinated Universal Time (UTC). User may change the time zone configuration by executing the following:

```
admin@nvos:~$ nv set system date-time timezone Etc/UTC
admin@nvos:~$ nv config apply
```

2.3.6.1.2 Clock

The system date and time can be manually changed. NTP servers configured on the switch will supersede any manually entered date-time settings.

```
admin@nvos:~$ nv action change system date-time 2024-12-24 10:25:13
```

2.3.6.2 NTP

NVOS supports Network Time Protocol (NTP) for synchronizing the time of the system.

2.3.6.2.1 NTP Servers from DHCP

By default, NTP will obtain the NTP servers from DHCP. It is possible to disable it by executing the following:

```
admin@nvos:~$ nv set system ntp dhcp disabled
admin@nvos:~$ nv config apply
```

2.3.6.2.2 Configure NTP Servers

A new NTP server can be added by executing the following:

```
admin@nvos:~$ nv set system ntp server 10.11.100.5
admin@nvos:~$ nv config apply
```

NTP can be configured to accept packets coming from an authenticated source only.
NTP authentication is disabled by default.
An authentication key may be created and used to authenticate incoming NTP packets. To ensure the key is utilized, the following conditions must be met:

- Enable NTP authentication

  ```
  admin@nvos:~$ nv set system ntp authentication enabled
  admin@nvos:~$ nv config apply
  ```

- Add a new NTP key, and associate it with the server:

  ```
  admin@nvos:~$ nv set system ntp key 15 value SECRET
  admin@nvos:~$ nv set system ntp key 15 trusted yes
  admin@nvos:~$ nv set system ntp server 10.11.100.5 key 15
  ```

```
admin@nvos:~$ nv config apply
```

- NTP server must use the same key.

NVOS support various attributes for NTP servers and keys, for more information please see NTP Commands.

2.3.7 Configure Syslog Servers

User can configure the system to send syslog logs to a centralized logging server (one or more):

```
admin@nvos:~$ nv set system syslog server 10.20.30.40
admin@nvos:~$ nv config apply
```

For more syslog configurations, please see Remote Logging section.

2.4 Test Cable Connectivity

By default, NVOS enables all data plane ports (every InfiniBand port). To view link status, run the `nv show interface` command or `nv show platform transceiver`.

When running NVUE commands to configure the switch, run the `nv config save` command before you reboot. The command saves the applied configuration to the startup configuration so that the changes persist after the reboot.

```
admin@nvos:~$ nv config save
```

2.5 Configure Connection-Mode In an InfiniBand Interface

The connection-mode attribute indicates to which peer this port is connected. This feature is dedicated to Q3200-RA system which can be connected to NDR systems (or older) and to XDR systems.
In case the port is connected to XDR systems, the connection-mode should be XDR. If it connected to NDR system, the connection-mode should be NDR. By default, all the ports are configured to connection-mode XDR.
When connecting port to NDR system, the port connection-mode needs to be changed with the set command (nv set interface link connection-mode) to 'ndr' in order to have link up with full functionality. Changing port connection-mode will move the port to down and up again when configuration ends.

3 Installation Management

This section describes how to manage, install, and upgrade NVOS.

3.1 Installing a New NVOS Image

The default password for the *admin* user account is **admin**.

It is recommended to change the default password when logging in for the first time. ONIE includes options that allow to change the default password for the *admin* account automatically when installing a new NVOS image. Refer to ONIE Installation Options.

A new NVOS image can be installed using ONIE—an open source project (equivalent to PXE on servers)—that enables the installation of network operating systems (NOS) on bare metal switches.

Before installing NVOS, the switch may be in one of the two following states:

1. The switch does not contain an image (the switch is only running ONIE).
2. NVOS is already on the switch, but NVUE commands are to be used to reinstall NVOS or upgrade to a newer version.

The sections below describe some of the different ways to install the NVOS image. Steps show how to install directly from ONIE (if no image is on the switch) and from NVOS (if the image is already on the switch). For additional methods to find and install the NVOS image, see the ONIE Design Specification.

To get into ONIE, you need to interrupt the GRUB countdown screen by pressing the "ESC" or "F4" key and choose the appropriate menu entry.

The NVOS image can be downloaded from the NVIDIA Enterprise Support Portal.

> Installing the NVOS image is destructive. Configuration files on the switch are not saved, so copy them to a different server before installation.

In the following procedures, the following is possible:

- The NVOS image can be named using any of the ONIE naming schemes mentioned here
- The `sudo onie-install -h` command can be run to show the ONIE installer options

3.1.1 Install Using a DHCP/Web Server With DHCP Options

To install NVOS using a DHCP or web server *with* DHCP options, set up a DHCP/web server on a laptop and connect the eth0 management port of the switch to the laptop. After connecting the cable, the installation proceeds as follows:

1. The switch boots up and requests an IP address (DHCP request).
2. The DHCP server acknowledges and responds with DHCP option 114 and the location of the installation image.
3. ONIE downloads the NVOS image, installs, and reboots.
   NVOS should now be running.

> The most common way is to send DHCP option 114 with the entire URL to the web server (this can be the same system). However, there are other ways to use DHCP even if you do not have full control over DHCP. See the ONIE user guide for information on partial installer URLs and advanced DHCP options—both articles list more supported DHCP options.

Example DHCP configuration with an ISC DHCP server:

```
subnet 172.0.24.0 netmask 255.255.255.0 {
  range 172.0.24.20 172.0.24.200;
  option default-url = "http://172.0.24.14/onie-installer-x86_64";
}
```

Example DHCP configuration with dnsmasq (static address assignment):

```
dhcp-host=sw4,192.168.100.14,6c:64:1a:00:03:ba,set:sw4
dhcp-option=tag:sw4,114,"http://roz.rtplab.test/onie-installer-x86_64"
```

If a web server is not accessible, this free Apache example can be used.

3.1.2 Install Using a DHCP/Web Server Without DHCP Options

Follow the steps below if logging into the switch on a serial console (ONIE), or log in on the console or with SSH (Install from NVOS).

Install from ONIE

1. Place the NVOS image in a directory on the web server.
2. Run the `onie-nos-install` command:

```
ONIE:/ #onie-nos-install http://10.0.1.251/path/to/nvos-amd64-25.01.0002.bin
```

3.1.3 Install Using a Web Server With no DHCP

Follow the steps below if logging into the switch on a serial console (ONIE), or log in on the console or with SSH (Install from NVOS) but *no* DHCP server is available.

> ⚠ A console connection is needed to access the switch. This procedure cannot be performed remotely.

Install from ONIE

1. ONIE is in discovery mode. Disable discovery mode with the following command:

```
onie# onie-discovery-stop
```

   On older ONIE versions, if the `onie-discovery-stop` command is not supported, run the following command:

```
onie# /etc/init.d/discover.sh stop
```

2. Assign a static address to eth0 with the `ip addr add` command:

```
ONIE:/ #ip addr add 10.0.1.252/24 dev eth0
```

3. Place the NVOS image in a directory on your web server.
4. Run the installer manually (because there are no DHCP options):

```
ONIE:/ #onie-nos-install http://10.0.1.251/path/to/nvos-install-x86_64.bin
```

3.1.4 Install Using FTP Without a Web Server

Follow the steps below if your laptop is on the same network as the switch eth0 interface but *no* DHCP server is available.

Install from ONIE

1. Set up DHCP or static addressing for eth0. The following example assigns a static address to eth0:

   ```
   ONIE:/ #ip addr add 10.0.1.252/24 dev eth0
   ```

2. If static addressing is being used, disable ONIE discovery mode:

   ```
   onie# onie-discovery-stop
   ```

   On older ONIE versions, if the `onie-discovery-stop` command is not supported, run the following:

   ```
   onie# /etc/init.d/discover.sh stop
   ```

3. Place the NVOS image into a TFTP or FTP directory.
4. If DHCP options are not being used, run one of the following commands (tftp for TFTP or ftp for FTP):

   ```
   ONIE# onie-nos-install ftp://local-ftp-server/nvos-amd64-25.01.0002.bin
   ONIE# onie-nos-install tftp://local-tftp-server/nvos-amd64-25.01.0002.bin
   ```

3.1.5 Install Using a Local File

Follow the steps below to install the NVOS image referencing a local file.

Install from ONIE

1. Set up DHCP or static addressing for eth0. The following example assigns a static address to eth0:

   ```
   ONIE:/ #ip addr add 10.0.1.252/24 dev eth0
   ```

2. If you are using static addressing, disable ONIE discovery mode.

   ```
   onie# onie-discovery-stop
   ```

   On older ONIE versions, if the `onie-discovery-stop` command is not supported, run the following:

```
onie# /etc/init.d/discover.sh stop
```

3. Use scp (secure copy protocol) to copy the NVOS image to the switch.
4. Run the installer manually from ONIE:

```
ONIE:/ #onie-nos-install /path/to/local/file/nvos-amd64-25.01.0002.bin
```

Install from NVOS

1. Fetch the NVOS image on the switch.

```
admin@nvos:~$ nv action fetch system image scp://<username>:<password>@<ip-address>/var/www/html/
<new_image>
```

```
admin@nvos:~$ nv show system image files
Available image file
-------------------------
nvos-amd64-25.01.0003.bin
```

2. From the NVOS command prompt, using NVUE command, install the new image and follow the instruction on the screen.

```
admin@nvos:~$ nv action install system image files nvos-amd64-25.01.0003.bin
Operation will reboot the system.
If you choose 'y', the system will install the image and reboot.
If you choose 'N', the operation will abort without installing the image and without rebooting the system.
Do you want to continue? [y/N]
```

3. After reboot, run **nv show system image** to review your images.

```
admin@nvos:~$ nv show system image
           operational
----------  -------------------
current     2
next        2
partition1
  build-id  nvos-25.02.1936
partition2
  build-id  nvos-25.02.2930-010
```

3.1.6 Installing using an image copied from the host machine

1. Copy the image from the host machine to the switch.

> ⚠ *Note: you must copy an image to the predefined directory: "/host/nos-images/"*

```
user@host:~$ scp <path-to-system-image> <switch-admin-username>@<switch-ip-address>:/host/nos-images/
<desired-name>.bin
```

2. To install the image login to the switch and perform steps 2-3 from the "Install from NVOS" section.

3.1.7 Install Using a USB Drive

Follow the steps below to install the NVOS image using a USB drive.

> ⚠ Tip: Installing NVOS using a USB drive is not scalable. DHCP can scale to hundreds of switch installs with zero manual input, unlike USB installs.

3.1.7.1 Prepare for USB Installation

1. Download the appropriate NVOS image for your platform.
2. From a computer, prepare your USB drive by formatting it using one of the supported formats: FAT32, vFAT, or EXT2.
   Optional: Prepare a USB drive inside NVOS
   a. Insert the USB drive into the USB port on the switch running NVOS and log in to the switch. Examine output from `cat /proc/partitions` and `sudo fdisk -l [device]` to determine the location of your USB drive. For example, `sudo fdisk -l /dev/sdb`.
   These instructions assume the USB drive is the `/dev/sdb` device, which is typical if inserting the USB drive after the machine is already booted. However, if the USB drive was inserted during the boot process, it is possible that the USB drive is the `/dev/sda` device. Make sure to modify the commands below to use the proper device for the USB drive.
   b. Create a new partition table on the USB drive. If the `parted` utility is not on the system, install it with `sudo -E apt-get install parted`.

```
sudo parted /dev/sdb mklabel msdos
```

   c. Create a new partition on the USB drive:

```
sudo parted /dev/sdb -a optimal mkpart primary 0% 100%
```

   d. Format the partition to your filesystem of choice using *one* of the examples below:

```
sudo mkfs.ext2 /dev/sdb1
sudo mkfs.msdos -F 32 /dev/sdb1
sudo mkfs.vfat /dev/sdb1
```

   To use `mkfs.msdos` or `mkfs.vfat`, installation of the `dosfstools` package from the Debian software repositories is needed, as they are not included by default.

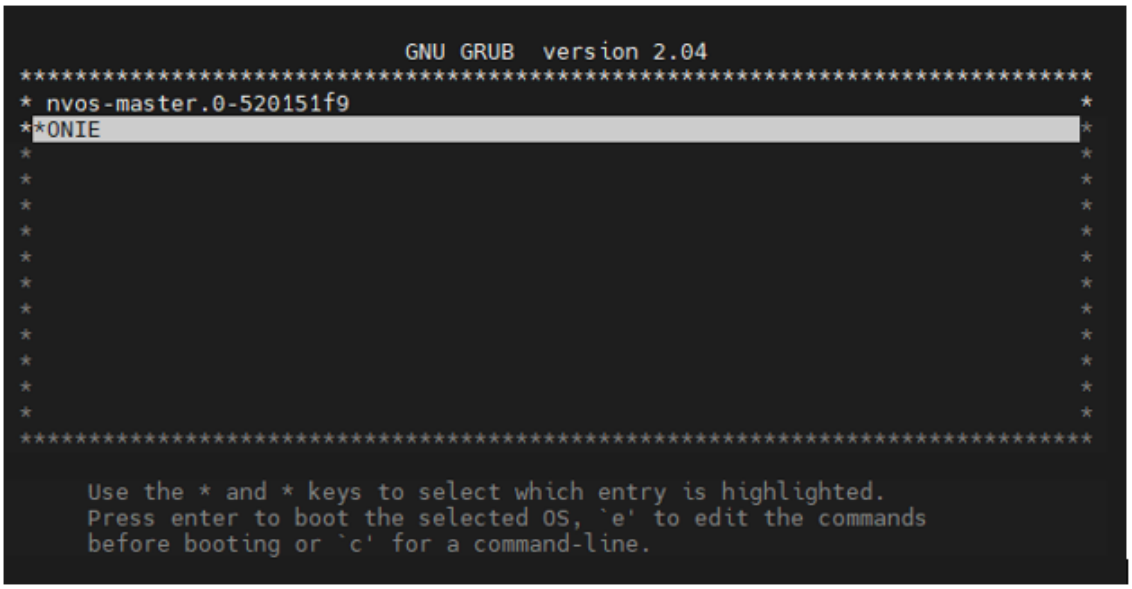
   e. To continue installing NVOS, mount the USB drive to move files:

```
sudo mkdir /mnt/usb
sudo mount /dev/sdb1 /mnt/usb
```

3. Copy the NVOS image to the USB drive, then rename the image file to `onie-installer-x86_64`.
   Any of the ONIE naming schemes mentioned here can also be used.
   When using a MAC or Windows computer to rename the installation file, the file extension can still be present. Make sure to remove the file extension so that ONIE can detect the file.

4. Insert the USB drive into the switch, then prepare the switch for installation:
    - If the switch is offline, connect to the console and power on the switch
    - If the switch is already online in ONIE, use the `reboot` command

   SSH sessions to the switch get dropped after this step. To complete the remaining instructions, connect to the console of the switch. NVOS switches display their boot process to the console; you need to monitor the console specifically to complete the next step.
5. Monitor the console and interrupt the GRUB countdown with the "ESC" or "F4" key when proposed, then select the ONIE option from the first GRUB screen shown below.



6. NVOS on x86 uses GRUB chainloading to present a second GRUB menu specific to the ONIE partition. No action is necessary in this menu to select the default option `ONIE: Install OS`.

7. The switch recognizes the USB drive and mounts it automatically. NVOS installation begins.
8. After installation completes, the switch automatically reboots into the newly-installed instance of NVOS.

3.1.8 Initial login

Once the NVOS image is successfully installed on the system, logging into the operating system becomes possible.

> Note: If you attempt to log in and execute commands or API calls immediately after the system boots, the NVUE service may not be fully loaded. The system will return the following message when it is still initializing:
>
> CLI:
>
> ```
> Error: NVOS CLI is unavailable
> System is initializing!
> This may take a few minutes
> ```
>
> RestAPI:
>
> ```
> {
> "type": "/errors/ServiceUnavailable",
> "title": "NVUE Service is unavailable",
> "status": 503,
> "detail": "System is initializing! This may take a few minutes."
> }
> ```

During the initial login to the NOS, a password change will be requested. This can be achieved by connecting to the switch via SSH and changing the password following the relevant message prompts.

> Once the password has been changed, please save your configuration in order to retain the password during subsequent reboots.
> If the switch is being managing only via ZTP, disregard the password change request.

3.1.9 Related Information

- ONIE Design Specification

3.2 Upgrading/Downgrading NVOS Image

The following pages provide information on upgrading and downgrading the operating system version on the device.

- Upgrading Operating System Software
- Deleting Unused Images
- Downgrading OS Software

3.2.1 Upgrading Operating System Software

3.2.1.1 Important Notes Before Upgrading OS Software

Consider the following items prior to upgrading the operating system:

- The system becomes unavailable while OS upgrade is in progress.
- The upgrade procedure burns the software image as well as the firmware.
- Before upgrading the software image, make sure to close all CLI sessions besides the one used to run the upgrade process.

3.2.1.2 In Service Software Upgrade (ISSU):

ISSU is a method to update the switch NOS software bundle, to a newer version then the currently running version, without disrupting the forwarding of traffic through the switch.

During the upgrade, the control and management planes incur unavailability while moving to the new OS version and refreshing the internal software databases.

However, the switch ASIC(s) are still forwarding traffic and keeping all physical and logical ports uninterrupted, even while the firmware inside the ASIC is being updated.

The ISSU flow involves interaction with the Subnet Manager (SM) to get its approval to launch ISSU.

The command to issue ISSU is the same command to perform regular software update with two additional flags

```
admin@nvos:~$ nv action install system image files new-nvos-image.bin ?

    [Enter]
    reboot                Install an image and reboot system
    force                 Force system reboot after system image installation
    issu                  Install image in ISSU mode
    skip-sm               Skip SM confirmation for ISSU
```

Additionally on systems that support ISSU, its status will be included in `nv show system image` output.

```
admin@nvos:~$ nv show system image
            operational          pending
----------  -------------------  -------
current     1
next        1
issu-status  no-issu
partition1
  build-id   nvos-25.02.2201
partition2
  build-id   nvos-25.02.2202
```

adding issu flag would trigger this flow and avoid the reboot.

skip-sm should be used only when there is a need to perform ISSU while SM is not available. This needs special care in a production environment.

ISSU is disabled by default on OpenSM, and needs a special configuration to enable it, please refer to OpenSM release notes.

3.2.1.3 Upgrading OS Software

To upgrade NVOS, perform the following steps.

1. Display the image (.bin file) that is currently available.

```
admin@nvos:~$ nv show system image
            operational          pending
----------  -------------------  -------
current     1
next        1
issu-status  no-issu
partition1
  build-id   nvos-25.02.2201
partition2
  build-id   nvos-25.02.2202
```

2. In case there are partition1 and partition2, Uninstall the image that isn't used as 'current' or 'next' prior to fetching the new image. Use the command **nv action uninstall system image,** for this purpose.

```
admin@nvos:~$ nv action uninstall system image
```

Another option is to clean-up all the unused images.

```
admin@nvos:~$ nv action uninstall system image force
```

3. Upload the new software image to the system using fetch command.

```
admin@nvos:~$ nv action fetch system image scp://<username>:<password>@<ip-address>/var/www/html/
<new_image>
```

4. Install the new image.

```
admin@nvos:~$ nv action install system image files new-nvos-image.bin
Operation will reboot the system.
If you choose 'y', the system will install the image and reboot.
If you choose 'N', the operation will abort without installing the image and without rebooting the system.
Do you want to continue? [y/N]
```

5. After reboot, display available images and verify that the new image now appears.

```
admin@nvos:~$ nv show system image
```

```
            operational          applied
----------  -------------------  -------
current     2
next        2
partition1
  build-id  <old_image>
partition2
  build-id  <new_image>
```

- After software reboot, the software upgrade will also automatically upgrade the firmware version.
- To recover from image corruption (e.g., due to power interruption), there are two installed images on the system. For more information, see the nv action boot-next system image command.

3.2.2 Deleting Unused Images

To delete unused images, conduct the following steps:

1. Get a list of the unused images.

```
admin@nvos:~$ nv show system image
            operational
----------  ---------------
current     1
next        1
issu-status  no-issu
partition1
  build-id   nvos-xx
```

2. Uninstall the unused images.

```
admin@nvos:~$ nv action uninstall system image
```

3.2.3 Downgrading OS Software

3.2.3.1 Prerequisites

Prior to downgrading software, please make sure the following prerequisites are met.

1. Log in to the via the CLI using the console port.
2. Backup configuration by conducting the following steps.
   a. copy commands.

```
admin@nvos:~$ nv config show
```

   b. Copy the output to a text file.

3.2.3.2 Downgrading Image

> The procedure described below assumes that booting and running is done from Partition 1 and the downgrade procedure is performed on Partition 2.

1. Log in to your system via the CLI as admin.
2. Display all installed images on the system.

```
 admin@nvos:~$ nv show system image
           operational          applied
----------  -------------------  -------
current     1
next        1
partition1
  build-id <image2>
```

3. Uninstall the image from partition 2 if it is installed. This step is required only if the user is trying to downgrade to an image that is already installed on the switch.

```
admin@nvos:~$ nv action uninstall system image <image>
```

4. Fetch the desired software image.

```
admin@nvos:~$ nv action fetch system image scp://<username>:<password>@<ip-address>/var/www/html/
<new_image>
```

5. Install the fetched image.

```
admin@nvos:~$ nv action install system image files new-nvos-image.bin force
```

6. After reboot, display available images and verify that the new image now appears.

```
admin@nvos:~$ nv show system image
           operational          applied
----------  -------------------  -------
current     1
next        1
partition1
  build-id <image1>
partition2
  build-id <image2>
```

> There are two installed images on the system. Therefore, if one of the images gets corrupted (due to power interruption, for example), in the next reboot the image will go up from the second partition.

3.2.3.3 Switching to Partition with Older Software Version

The system saves a backup configuration file when upgrading from an older software version to a newer one. If the system returns to the older software partition, it uses this backup configuration file.

> All configuration changes done with the new software are lost when returning to the older software version.

There are 2 instances where the backup configuration file does not exist:

- The user has run “reset factory” command, which clears all configuration files in the system
- The user has run “system config files” to a configuration file with different name then the backup file

> ⚠ Note that the configuration file becomes empty if the system is downgraded to a software version which has never been installed yet.

To allow switching partition to the older software version for the 2 aforementioned cases only, follow the steps below.

1. Set the boot partition.

```
admin@nvos:~$ nv action boot-next system image <next partition>
```

2. Save the configuration.

```
admin@nvos:~$ nv config save
```

3.3 Upgrading System Firmware

NVOS provides the ability to upgrade the Firmware of system components manually.

3.3.1 Upgrade ASIC Firmware

NVOS is bundled with a default ASIC firmware version. When updating the operating system software to a new version, an automatic firmware update process will be attempted by NVOS. The user can choose to override the default version.
To perform an automatic firmware update by the OS for a different switch firmware version without changing the OS version, import and install the firmware package as described below. The OS sets it as the new default firmware and performs the firmware update automatically.

3.3.1.1 ASIC Firmware Change

1. Display the ASIC firmware that is currently available.

```
admin@nvos:~$ nv show platform firmware ASIC
                operational          applied
--------------  -----------------    -------
part-number     MQM9700-NS2X-NS_Ax
actual-firmware 31.2014.0946
auto-update     enabled              enabled
fw-source       default              default
```

2. Import the firmware image (.mfa file) to the switch.

```
admin@nvos:~$ nv action fetch platform firmware ASIC /path/to/fw-image.mfa
```

Alternatively, you can upload the FW file from the host to the switch:

> ⚠ The firmware file must be copied to the predefined directory: "/host/fw-images/asic"

```
user@host:~$ scp <path-to-fw-file> <switch-admin-username>@<switch-ip-address>:/host/fw-images/asic/
<desired-name>.mfa
```

3. Configure default firmware source from user.

```
admin@nvos:~$ nv set platform firmware ASIC fw-source custom
```

4. Display system firmware component information.

```
admin@nvos:~$ nv show platform firmware ASIC
                  operational          applied
---------------   ------------------   -------
part-number       MQM9700-NS2X-NS_Ax
actual-firmware   31.2014.0946
auto-update       enabled              enabled
fw-source         default              default
```

5. Apply the configuration

```
admin@nvos:~$ nv config apply
```

6. Save the configuration.

```
admin@nvos:~$ nv config save
```

7. Install the firmware image.

```
admin@nvos:~$ nv action install platform firmware ASIC files fw-image.mfa
The operation will initiate a component firmware update.
Type [y] to install the firmware and reboot afterwards.
Type [N] to install the firmware without reboot.

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

8. Press 'Y' will cause system reboot to install and activate firmware.
9. Press 'N' requires manual reboot from user later on.

```
admin@nvos:~$ nv action reboot system
```

3.3.2 Upgrade Components Firmware

This section provides step-by-step instructions to manually check and update software and firmware on InfiniBand switch to ensure the system is up to date with the latest software and firmware versions.

> ⚠
> - These updates are not done every release. See NVIDIA NVOS Release Notes to see which versions should be used.
> - The upgrade process will require maintenance window.
> - If necessary, retrieve logs for customer support using the command "nv action generate system tech-support".

3.3.2.1 Component Image Update Using NVOS CLI

Firmware updates can be done by NVOS CLI commands. CLI commands are blocking, meaning each command must be finished before another one can be

There are two stages to upgrade each component:

1. Fetching a file from the unpacked bundle.

```
admin@nvos:~$ nv action fetch platform firmware <component-id> <remote-url>
```

   For details, see nv action fetch platform firmware.

2. Installing a component:

```
admin@nvos:~$ nv action install platform firmware <component-id> files <file-name>
```

   For details, see nv action install platform firmware files.

- <component-id> can be one of the following: ASIC, BIOS, and CPLD1.
- Once upgrading a specific CPLD, all other CPLDs will be upgraded as well.
- CPLD firmware bundle should include two different files: burn and refresh.

1. After installation, the switch will reboot automatically if the "skip-reboot" flag was not set.

   Recommended flow for CPLD upgrade:

   a. Install burn file with skip-reboot flag:

```
admin@nvos:~$ nv action install platform firmware CPLD1 files burn.vme skip-reboot
```

   b. Install refresh file:

```
admin@nvos:~$ nv action install platform firmware CPLD1 files refresh.vme
```

   The system will perform a power cycle automatically.

2. To verify firmware versions after system boot, run the following:

```
admin@nvos:~$ nv show platform firmware
```

   For details, see nv show platform firmware.

3.3.2.1.1 Transceiver Firmware Upgrade

This section is relevant only for systems that support external transceivers.

Firmware updates can be done by NVOS CLI commands. CLI commands are blocking, meaning each command must be finished before another one can be.

There are two stages to upgrade each component:

1. Fetching a file from the unpacked bundle.

```
admin@nvos:~$ nv action fetch platform firmware transceiver <file-path>
```

   For details, see nv action fetch platform firmware.

2. Installing transceiver firmware.

```
admin@nvos:~$ nv action install platform transceiver <transceiver-id> firmware files <file-name>
```

   For details, see nv action install platform transceiver firmware files.

The upgrade time can be reduced by using a utility script to burn transceiver firmware for all connected cables:

```
admin@nvos:~$ burn_all_modules <file-name>
```

In order to activate the transceiver firmware, NVOS will reset the transceiver as part of the install action.

To verify firmware version, run the following:

```
admin@nvos:~$ nv show platform transceiver <transceiver-id> firmware
```

For details, see nv show platform transceiver firmware.

3.3.2.2 Component Image Update Using RestAPI

RestAPI can be used from remote server to perform operations on the switch.
RestAPI is not blocking, meaning command can be sent before the previous finished. To deal with this nature, each command returns Task ID, use the Task ID to query for the result between the commands. State of “action_success” means the operation ended successfully.

Upgrades consist of fetch, install, and power cycle at the end of the entire process.

3.3.2.2.1 REST API Commands

Query command, should be executed between commands:

```
admin@nvos:~$ curl -k --user <nvos-user>:<nvos-password> --request GET 'https://<switch-ip>/nvue_v1/action/<task-id>'
```

- Fetching component image file:

```
admin@nvos:~$ curl -k --user <nvos-user>:<nvos-password> --request POST 'https://<switch-ip>/nvue_v1/platform/firmware/<component>' -H 'Content-Type: application/json' -d '{"@fetch": {"state": "start", "parameters": {"remote-url": "scp://<server-user>:<<server-password> >@<PATH_TO_FILE>"}}}'
```

- Install the component file:

```
admin@nvos:~$ curl -k --user <nvos-user>:<nvos-password> --request POST 'https://<switch-ip>/nvue_v1/
platform/firmware/<component>/files/</<file-name>' -H 'Content-Type: application/json' -d '{"@install":
{"state": "start", "parameters": {"force": false}}}'
```

- Fetching single image file:

```
admin@nvos:~$ curl -k --user <nvos-user>:<nvos-password> --request POST 'https://<switch-ip>/nvue_v1/
platform/firmware' -H 'Content-Type: application/json' -d '{"@fetch": {"state": "start", "parameters":
{"remote-url": "scp://<server-user>:<<server-password> >@<PATH_TO_FILE>"}}}'
```

- Install the single image file:

```
admin@nvos:~$ curl -k --user <nvos-user>:<nvos-password> --request POST 'https://<switch-ip>/nvue_v1/
platform/firmware/files/</<file-name>' -H 'Content-Type: application/json' -d '{"@install": {"state":
"start", "parameters": {"force": false}}}'
```

- After system boot, check firmware version:

```
admin@nvos:~$ curl -k --user <nvos-user>:<nvos-password> --request GET 'https://<nvos-ip>/nvue_v1/platform/
firmware'
```

3.3.2.3 Error Status Catalog

Use the table below to identify the errors and their meaning.

Bundles List

BIOS	
Scenario	**Error**
Selected file for installation doesn't exist	Failed to install BIOS firmware file: No such firmware
Bad or corrupted file	Invalid file: /host/fw-images/bios/bad_file.cab
Bad Onie version	ERROR: ONIE {} or later is required
Failed to enable ONIE firmware update mode	ERROR: failed to enable ONIE firmware update mode
Failed to disable ONIE firmware update mode	ERROR: failed to disable ONIE firmware update mode
Installation script was interrupted by signal	WARNING: Interrupted by ${_sig}: disable ONIE firmware update mode
CPLD	
Scenario	**Error**
Selected file for installation doesn't exist	Failed to install CPLD firmware file: No such firmware
Bad or corrupted file	Invalid file: /host/fw-images/cpld/bad_file.vme
MST service not started (never started or failed to start)	ERROR: mst driver is not loaded
MST device path doesn't exist or failed to it	ERROR: Failed to get mst device: pattern={}, devices={}

CPLD update command failed	ERROR: Failed to update {} firmware: {}

3.4 Installation Management Commands

For examples of how to use each REST API command type, see the following sections:

- GET: View a Configuration
- PATCH: Make a Configuration Change
- POST: Action Operations
- DELETE: Unset a Configuration Change

3.4.1 Version

3.4.1.1 nv show system version

	nv show system version Displays system version information, including 'base-os' and 'product-release'.
Syntax Description	N/A

Default	N/A
History	25.02.2002 25.02.5002 Updated output
Example	admin@nvos:~$ nv show system version operational ---------- ---------- kernel 5.10.0-30-2-amd64 build-date Wed Jun 18 15:19:56 UTC 2025 image nvos-25.02.4014 onie 2025.05-5.3.0017-115200
REST API	GET https://<ip>/nvue_v1/system/version
Related Commands	nv show system nv show system image
Notes	

3.4.2 Image

3.4.2.1 nv show system image

	nv show system image Show system image information. Display version information related to system image.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos: ~$ nv show system image operational ---------- ---------- current 1 next 1 issu-status no-issu partition1 build-id nvos-25.02.4014
REST API	GET https://<ip>/nvue_v1/system/image
Related commands	nv show system image files
Notes	

3.4.2.2 nv show system version image

	nv show system version image Displays information about the system's boot image.
Syntax Description	N/A
Default	N/A
History	25.02.5002

Example	admin@nvos: ~$ nv show system version image operational ---------- ---------- build-id nvos-25.02.2988-001 build-date Wed Mar 5 11:23:22 UTC 2025
REST API	GET https://<ip>/nvue_v1/system/version/image
Related commands	
Notes	

3.4.2.3 nv show system version packages installed

	nv show system version packages installed Lists all currently installed software packages on the system.
Syntax Description	N/A
Default	N/A
History	25.02.5002
Example	admin@nvos: ~$ nv show system version packages installed Installed Software Description Package Version ---------- ---------- -------- ------- acpi displays information on ACPI devices acpi 1.7-1.2
REST API	GET https://<ip>/nvue_v1/system/version/packages/installed
Related commands	
Notes	

3.4.2.4 nv show system version packages installed id

	nv show system version packages installed <installed-id> Displays detailed information about a specific installed software package. Replace with the actual package name.
Syntax Description	N/A
Default	N/A
History	25.02.5002
Example	admin@nvos: ~$ nv show system version packages installed acl operational ---------- ---------- package acl version 2.2.53-10 description access control list - utilities
REST API	GET https://<ip>/nvue_v1/system/version/packages/installed/<installed-id>
Related commands	
Notes	

3.4.2.5 nv show system image files

	nv show system image files Show files in the image directory on the switch file system.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show system image files Available image file --------------------------- nvos-amd64-25.01.2434-013.bin nvos-amd64-25.01.2504.bin
REST API	GET https://<ip>/nvue_v1/system/image/files
Related commands	nv show system image
Notes	

3.4.2.6 nv action fetch system image

	nv action fetch system image <remote-url> Fetch/download a file from a remote server and stores it locally.	
Syntax Description	remote-url	• Remote url path to fetch file from. • Format: [protocol]://username[:password]@hostname/path/filename • Supported protocols: SCP, HTTPS, FILE, FTP, and SFTP. • The password must be encoded if it contains special characters and is provided as part of the command.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action fetch system image scp://my_user:my_password@server/path/image.bin admin@nvos:~$ nv action fetch system image https://some.domain.com/path/image.bin admin@nvos:~$ nv action fetch system image file:///path/image.bin	
REST API	POST https://<ip>/nvue_v1/system/image	
Related Commands	nv show system image files	
Notes	Using https requires the remote server to have a valid CA certificate.	

3.4.2.7 nv action install system image files

	nv action install system image files {image-file} [force] [reboot <yes/no>] Install system image from a binary file and reboot the system.	
Syntax Description	image-file	Path to the binary file to install on the OS filesystems
	force	Force the action without asking for user confirmation
	reboot	Install image with or without system reboot

Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv action install system image files nvos.bin The operation will install the image and initiate a reboot. Type [y] to install the image and reboot. Type [N] to abort. Do you want to continue? [y/N] N Image install aborted by user
REST API	POST https://<ip>/nvue_v1/system/image/files/
Related commands	nv show system image
Notes	Executing the command for an image that is already installed on the other partition will not install the image again, it will only change the "boot-next" image.

3.4.2.8 nv action rename system image files

	nv action rename system image files <image> <new-name> Rename an image file.	
Syntax Description	image	Source image file name
	new-name	Destination image file name
Default	N/A	In case of an empty parameter performs a regular reboot without force.
History	25.02.2002	
Example	admin@nvos:~$ nv action rename system image files old_name.img new_name.img	
REST API	POST https://<ip>/nvue_v1/system/image/files/<file-name>	
Related Commands	nv show system image files	
Notes		

3.4.2.9 nv action upload system image files

	nv action upload system image files <image> <remote-url> Upload an image file to remote location.	
Syntax Description	image	Image file to upload
	remote-url	Destination image file name Remote url path to upload a file to. Format: [protocol]://username[:password]@hostname/path/filename Supported protocols: SCP, FTP, SFTP, and HTTPS
Default	N/A	
History	25.02.2002 25.02.4002 Added HTTPS support in remote-url	

Example	admin@nvos:~$ nv action upload system image image_file.bin scp://my_user:my_password@server/path/image_file.bin
REST API	POST https://<ip>/nvue_v1/system/image/files/<file-name>
Related commands	nv show system image files
Notes	

3.4.2.10 nv action uninstall system image

	nv action uninstall system image Removes old unused images that are not current or boot-next.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv action uninstall system image Cleaning up old unused images... Removing image nvos-old Images cleanup done Action succeeded admin@nvos:~$ nv action uninstall system image Cleaning up old unused images... No image(s) to cleanup Action succeeded
REST API	POST https://<ip>/nvue_v1/system/image
Related commands	nv show system image
Notes	

3.4.2.11 nv action delete system image files

	nv action delete system image files <image> Delete an image from file system.	
Syntax Description	image	Image file name
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action delete system image files nvos.bin	
REST API	POST https://<ip>/nvue_v1/system/image/files/<file-name>	
Related Commands	nv show system image files	
Notes		

3.4.2.12 nv action uninstall system image force

	nv action uninstall system image force Uninstall the system image. Uninstall old unused images that are not current image.	
Syntax Description	image-id	System image identifier
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv action uninstall system image force
	Uninstalling system image: nvos
	Image nvos uninstalled successfully
	Action succeeded
	
	admin@nvos:~$ nv action uninstall system image
	Uninstalling system image: nvos
	Action failed with the following issue:
	  Image nvos can't be uninstalled. Image does not exist</pre>	

REST API	POST https://<ip>/nvue_v1/system/image	
Related commands	nv show system image	
Notes	Image ID can be retrieved by the "nv show system image" command.	

3.4.2.13 nv action boot-next system image

	nv action boot-next system image <image-id> Set boot-next system image. Set image to boot from at the next boot.	
Syntax Description	image-id	System image identifier
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv action boot-next system image nvos
	Setting system image nvos as boot-next..
	Image nvos set as boot-next
	Action succeeded
	
	admin@nvos:~$ nv action boot-next system image nvos
	Setting system image nvos as boot-next..
	Action failed with the following issue:
	  Failed to set boot-next: Image nvos does not exist</pre>	

REST API	POST https://<ip>/nvue_v1/system/image	
Related commands	nv show system image	
Notes	• Image ID can be retrieved by "nv show system image" command. • Select which partition to boot from. • Booting to an image with existing DB (with an image that is not freshly installed), will not migrate the configurations from the current image.	

3.4.3 Firmware

3.4.3.1 nv show platform firmware

	nv show platform firmware Show platform firmware information. Show firmware information for platform components like ASIC, BIOS, SSD, CPLD, and transceiver.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~# nv show platform firmware Name Actual FW Part Number FW Source ---------- ---------- ---------- --------- ASIC 31_2012_2906-032 MQM9700-NS2X-NS_Ax default BIOS 0ACQF_06.01.002 N/A N/A CPLD1 CPLD000232_REV0700 N/A N/A CPLD2 CPLD000324_REV0300 N/A N/A CPLD3 CPLD000000_REV0000 N/A N/A SSD 0202-002 StorFly VSFBM4XC016G-MLX2 N/A transceiver N/A N/A N/A
REST API	GET https://<ip>/nvue_v1/platform/firmware
Related commands	
Notes	

3.4.3.2 nv show platform firmware id

	nv show platform firmware <component-id> Show platform firmware information for specific component. Show firmware information for platform components such as ASIC, BIOS, CPLD, SSD, and transceiver.	
Syntax Description	component-id	Platform component name (e.g., ASIC, BIOS, transceiver)
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show platform firmware ASIC operational applied ---------- ---------- ------- part-number MQM9700-NS2X-NS_Ax actual-firmware 31_2012_2906-032 auto-update enabled fw-source default	
REST API	GET https://<ip>/nvue_v1/platform/firmware/<component-id>	
Related Commands	nv show platform firmware	
Notes		

3.4.3.3 nv show platform firmware files

	nv show platform firmware <component-id> filesDisplay the list of available firmware files for specific component. Display the list of available firmware files for platform components such as ASIC, BIOS, CPLD and transceiver.	
Syntax Description	component-id	Platform component name (e.g., ASIC, BIOS, transceiver)
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show platform firmware ASIC files
	Available firmware files File path
	---------- ----------
	fw-QTM2-rel-31_2012_3008-EVB.mfa /host/fw-images/fw-QTM2-rel-31_2012_3008-EVB.mfa
	
	admin@nvos:~$ nv show platform firmware transceiver files
	Available Firmware Files File Path
	---------- ----------
	fw_47_150_03003_dev_signed.bin /host/fw-images/module/fw_47_150_03003_dev_signed.bin</pre>	

REST API	GET https://<ip>/nvue_v1/platform/firmware/{component-id}/files	
Related commands	nv show platform firmware nv action install platform firmware <component-id> files <file-name> nv action install platform transceiver <transceiver-id> firmware files <file-name>	
Notes		

3.4.3.4 nv set/unset platform firmware

	nv set platform firmware [ASIC | BIOS] [auto-update {enabled,disabled} | fw-source {default,custom}] nv unset platform firmware [ASIC | BIOS] [auto-update {enabled,disabled} | fw-source {default,custom}] Update the platform ASIC or BIOS component configuration.	
Syntax Description	auto-update	Firmware component auto-update state
	fw-source	Firmware component default source
Default	auto-update	ASIC: enabled BIOS: disabled
	fw-source	default
History	25.02.2002 25.02.6007 Updated auto-update BIOS default to "disabled"	

Example	<pre>admin@nvos:~$ nv set platform firmware ASIC fw-source custom
	admin@nvos:~$ nv set platform firmware ASIC auto-update disabled</pre>	

REST API	PATCH https://<ip>/nvue_v1/platform/firmware/ASIC	
Related commands		
Notes	The configuration on BIOS is not available on system with BMC.	

3.4.3.5 nv action install platform firmware files

	nv action install platform firmware <platform-component-id> files <file-name> [force] [skip-reboot]Install platform firmware image. ASIC Image will be installed to stage location and will be installed at the next reboot time.	
Syntax Description	platform-component-id	Platform component name (e.g., ASIC, BIOS, transceiver, SSD)
	file-name	Path to the firmware file.
	force	Force the action without asking for user confirmation.
	skip-reboot	Skip reboot after firmware installation. If force was specified, the reboot will still be skipped.
Default	N/A	
History	25.02.2002	
Example	 admin@nvos:~$ nv action install platform firmware ASIC files/tmp/fw-QTM2.mfa Installing firmware: /tmp/fw-QTM2.mfa Firmware fw-QTM2.mfa successfully installed Action succeeded	
REST API	POST https://<ip>/nvue_v1/platform/firmware/{platform-component-id}/files/{file-name}	
Related commands	nv set/unset platform firmware ASIC auto-update nv action reboot system	
Notes	• ASIC Image will be installed to stage location and will be installed at the next reboot time • Installation respects "auto-update" and "fw-source" configuration • To install firmware on a transceiver, use the following command: nv action install platform transceiver firmware files • After installing firmware, initiating system reboot via the "nv action reboot system" command will perform a power-cycle to load the new firmware. • The percentage progress bar displayed during single-platform firmware installation reflects the overall installation progress, not the progress of an individual component.	

3.4.3.6 nv action delete platform firmware files

	nv action delete platform firmware <component-id> files <file-name>Delete firmware file from the filesystem for specific component such as ASIC, BIOS, CPLD and transceiver.	
Syntax Description	file-name	firmware file name
	component-id	Platform component name (e.g., ASIC, BIOS, transceiver)
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv action delete platform firmware ASIC files fw-test.mfa File delete successfully Action succeeded admin@nvos:~$ nv action delete platform firmware transceiver files fw-test.bin File delete successfully Action succeeded
REST API	POST https://<ip>/nvue_v1/platform/firmware/{component-id}/files/{file-name}
Related commands	nv show platform firmware <component-id> files nv action fetch platform firmware <component-id> <url> nv action upload platform firmware <component-id> files <file-name> <remote-url> nv action rename platform firmware <component-id> files <file-name> <new-name> nv action install platform firmware <component-id> files <file-name> [force] nv show platform transceiver <transceiver-id> firmware nv action install platform transceiver <transceiver-id> firmware files
Notes	

3.4.3.7 nv action rename platform firmware files

	nv action rename platform firmware <component-id> files <file-name> <new-name>Rename available firmware file for platform components such as ASIC, BIOS, CPLD and transceiver.	
Syntax Description	file-name	Firmware file name
	new-name	New name for firmware file
	component-id	Platform component name (e.g., ASIC, BIOS, transceiver)
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action rename platform firmware ASIC files fw-QTM.mfa new-fw-QTM.mfa File renamed successfully Action succeeded admin@nvos:~$ nv action rename platform firmware transceiver files fw-test.bin new-fw-test.bin File renamed successfully Action succeeded	
REST API	POST https://<ip>/nvue_v1/platform/firmware/{component-id}/files/{file-name}	
Related commands	nv show platform firmware <component-id> files nv action fetch platform firmware <component-id> <url> nv action upload platform firmware <component-id> files <file-name> <remote-url> nv action delete platform firmware <component-id> files <file-name> nv action install platform firmware <component-id> files <file-name> [force] nv show platform transceiver <transceiver-id> firmware nv action install platform transceiver <transceiver-id> firmware files	
Notes		

3.4.3.8 nv action fetch platform firmware

	nv action fetch platform firmware <component-id> <remote-url>Fetch remote firmware file to local filesystem for platform components such as ASIC, BIOS, CPLD and transceiver.

Syntax Description	component-id	Platform component name (e.g., ASIC, BIOS, transceiver)
	remote-url	• Remote url path to fetch file from. • Format: [protocol]://username[:password]@hostname/path/filename • Supported protocols: SCP, HTTPS, FILE, FTP, and SFTP. • The password must be encoded if it contains special characters and is provided as part of the command.
Default	N/A	
History	25.02.2002	
Example	 admin@nvos:~$ nv action fetch platform firmware ASIC scp://my_user:my_password@hostname/dir/file.mfa Fetching file … File fetched successfully Action succeeded admin@nvos:~$ nv action fetch platform firmware transceiver scp://my_user:my_password@hostname/dir/file.bin Fetching file … File fetched successfully Action succeeded admin@nvos:~$ nv action fetch platform firmware ASIC file:///dir/file.mfa Fetching file … File fetched successfully Action succeeded	
REST API	POST https://<ip>/nvue_v1/platform/firmware/{component-id}/files/{file-name}	
Related commands	nv show platform firmware <component-id> files nv action rename platform firmware <component-id> files <file-name> <new-name> nv action upload platform firmware <component-id> files <file-name> <remote-url> nv action delete platform firmware <component-id> files <file-name> nv action install platform firmware <component-id> files <file-name> [force] nv show platform transceiver <transceiver-id> firmware nv action install platform transceiver <transceiver-id> firmware files	
Notes		

3.4.3.9 nv action upload platform firmware files

	nv action upload platform firmware <component-id> files <file-name> <remote-url>Upload available firmware file for platform components such as ASIC, BIOS, CPLD and transceiver to remote server.	
Syntax Description	component-id	Platform component name (e.g., ASIC, BIOS, transceiver)
	file-name	Firmware file name
	remote-url	Destination image file name Remote url path to upload a file to. Format: [protocol]://username[:password]@hostname/path/filename Supported protocols: SCP, FTP, SFTP, and HTTPS
Default	N/A	
History	25.02.2002 25.02.4002 Added HTTPS support in remote-url	

Example	admin@nvos:~$ nv action upload platform firmware ASIC files fw-QTM.mfa new-fw-QTM.mfa Action executing ... File upload successfully Action succeeded
REST API	POST https://<ip>/nvue_v1/platform/firmware/{component-id}/files/{file-name}
Related commands	nv show platform firmware <component-id> files nv action fetch platform firmware <component-id> <url> nv action delete platform firmware <component-id> files <file-name> nv action install platform firmware <component-id> files <file-name> [force] nv show platform transceiver <transceiver-id> firmware nv action install platform transceiver <transceiver-id> firmware files
Notes	

4 NVIDIA User Experience (NVUE)

NVUE is an object-oriented, schema driven model of a complete NVOS system (hardware and software) providing a robust API that allows for multiple interfaces to both view (show) and configure (set and unset) any element within a system running the NVUE software.

4.1 NVUE Object Model

The NVUE object model definition uses the OpenAPI specification (OAS). Similar to YANG (RFC 6020 and RFC 7950), OAS is a data definition, manipulation, and modeling language (DML) that lets you build model-driven interfaces for both humans and machines. Although the computer networking and telecommunications industry commonly uses YANG (standardized by IETF) as a DML, the adoption of OpenAPI is broader, spanning cloud to compute to storage to IoT and even social media. The OpenAPI Initiative (OAI) consortium leads OpenAPI standardization, a chartered project under the Linux Foundation.

The OAS schema forms the management plane model with which you configure, monitor, and manage the NVOS switch. The v3.0.2 version of OAS defines the NVUE data model.

Like other systems that use OpenAPI, the NVUE OAS schema defines the endpoints (paths) exposed as RESTful APIs. With these REST APIs, you can perform various create, retrieve, update, delete, and eXecute (CRUDX) operations. The OAS schema also describes the API inputs and outputs (data models).

You can use the NVUE object model in these two ways:

- Through the NVUE REST API, where you run the GET, PATCH, DELETE, and other REST APIs on the NVUE object model endpoints to configure, monitor, and manage the switch. Because of the large user community and maturity of OAS, you can use several popular tools and libraries to create client-side bindings to use the NVUE REST API.
- Through the NVUE CLI, where you configure, monitor and manage the NVOS network elements. The CLI commands translate to their equivalent REST APIs, which NVOS then runs on the NVUE object model.

The CLI and the REST API are equivalent in functionality; you can run all management operations from the REST API or the CLI. The NVUE object model drives both the REST API and the CLI management operations. All operations are consistent; for example, the CLI `nv show` commands reflect any PATCH operation (create) you run through the REST API.

NVUE follows a declarative model, removing context-specific commands and settings. It is structured as a *big tree* that represents the entire state of a NVOS instance. At the base of the tree are high level branches representing objects, such as *system* and *interface*. Under each of these branches are further branches. As you navigate through the tree, you gain a more specific context. At the leaves of the tree are actual attributes, represented as key-value pairs. The path through the tree is similar to a filesystem path.

Please note that configuration with dependencies should be applied in two different steps.

NVOS installs NVUE by default and enables the NVUE service `nvued`.

4.2 NVUE CLI Overview

- 4.2.1 Command Syntax
- 4.2.2 Command Completion
- 4.2.3 Command Question Mark
- 4.2.4 Command Abbreviation
- 4.2.5 Command Help
- 4.2.6 Command List
- 4.2.7 Command History
- 4.2.8 Command Ranges
- 4.2.9 Command Categories
    - 4.2.9.1 Configuration Commands
    - 4.2.9.2 Monitoring Commands

The NVUE CLI has a flat structure as opposed to a modal structure. This means that you can run all commands from the primary prompt instead of only in a specific mode.

4.2.1 Command Syntax

NVUE commands all begin with `nv` and fall into one of four syntax categories:

- Configuration (`nv set` and `nv unset`)
- Monitoring (`nv show`)
- Action commands (`nv action`)
- Configuration management (`nv config`).

4.2.2 Command Completion

As you enter commands, you can get help with the valid keywords or options using the Tab key. For example, using Tab completion with `nv set` displays the possible options for the command and returns you to the command prompt to complete the command.

```
admin@nvos:~$ nv set <<press Tab>>
acl        interface  platform   system
```

4.2.3 Command Question Mark

You can type a question mark (`?`) after a command to display required information quickly and concisely. When you type `?` , NVUE specifies the value type, range, and options with a brief description of each; for example:

```
admin@nvos:~$ nv set system config auto-save ?

    [Enter]
    state                 Enable/Disable configuration automatic save feature
```

NVUE also indicates if you need to provide specific values for the command:

```
admin@nvos:~$ nv set interface ?
    <interface-id>          Interface (interface-name)
```

4.2.4 Command Abbreviation

NVUE supports command abbreviation, where you can type a certain number of characters instead of a whole command to speed up CLI interaction. For example, instead of typing **nv show interface**, you can type **nv sh int**.

If the command you type is ambiguous, NVUE shows the reason for the ambiguity so that you can correct the shortcut. For example:

```
admin@nvos:~$ nv s i
Ambiguous Command:
   set interface
   show interface
```

4.2.5 Command Help

As you enter commands, you can get help with command syntax by entering **-h** or **--help** at various points within a command entry. For example, to examine the options available for **nv set interface**, enter **nv set interface -h** or **nv set interface --help**.

```
admin@nvos:~$ nv set interface -h
usage:
  nv [options] set interface <interface-id>

Description:
  interface             Update all interfaces. Provide single interface or multiple interfaces using ranging (e.g.
sw1-2p1-2 -> sw1p1,sw1p2,sw2p1,sw2p2).

Identifiers:
  <interface-id>        Interface (interface-name)

General Options:
  -h, --help            Show help.
```

4.2.6 Command List

You can list all the NVUE commands by running **nv list-commands**.

4.2.7 Command History

At the command prompt, press the Up Arrow and Down Arrow keys to move back and forth through the list of commands you entered. When you find a given command, you can run the command by pressing Enter. Optionally, you can modify the command before you run it.

4.2.8 Command Ranges

A single interface or multiple interfaces can be provided using a range such as sw1-2p1-2 -> sw1p1,sw1p2,sw2p1,sw2p2.

Example:

```
admin@nvos:~$ nv set interface sw1p1s1,sw1p1s2 description TEST
admin@nvos:~$ nv config diff
- set:
    interface:
      sw1p1s1-2:
        description: TEST
        type: nvl
```

```
admin@nvos:~$  nv set interface sw1p1-2s1 description TEST
admin@nvos:~$  nv config diff
- set:
    interface:
      sw1p1-2s1,sw2p1-2s1:
        description: TEST
        type: nvl
```

4.2.9 Command Categories

The NVUE CLI has a flat structure; however, the commands are in four functional categories:

- Configuration
- Monitoring
- Action
- Configuration Management

4.2.9.1 Configuration Commands

The NVUE configuration commands modify switch configuration. You can set and unset configuration options.

The `nv set` and `nv unset` commands are in the following categories. Each command group includes subcommands. Use command completion (press the Tab key) to list the subcommands.

Command Group	Description
`nv set interface <interface-id>` `nv unset interface <interface-id>`	Configures the switch interfaces. Use this command to configure eth0/eth1/lo/nvl interfaces.
`nv set system` `nv unset system`	Configures the hostname of the switch, pre and post login messages, log level, users, password-hardening policies etc.
`nv set acl` `nv unset acl`	Configures ACLs.
`nv set platform` `nv unset platform`	Configures the platform firmware configuration.

4.2.9.2 Monitoring Commands

The NVUE monitoring commands show various parts of the network configuration. For example, you can show the complete network configuration or only interface configuration. The monitoring commands are in the following categories. Each command group includes subcommands. Use command completion (press the Tab key) to list the subcommands.

Command Group	Description
`nv show acl`	Shows Access Control List configuration.
`nv show action`	Shows information about the action commands job.
`nv show interface`	Shows interface configuration.
`nv show platform`	Shows platform configuration, such as hardware and software components.
`nv show system`	Shows global system settings, such as the images, tech-support files and log . You can also see system login messages and switch reboot history.
`nv show vrf`	Shows VRF configuration.
`nv show ib`	Shows system and MGMT GUIDs and LIDs

If there are no pending or applied configuration changes, the `nv show` command only shows the running configuration (under operational).

Additional options are available for the `nv show` commands. For example, you can choose the configuration you want to show (pending, applied, startup, or operational). You can also turn on colored output, and paginate specific output.

Option	Description
`--applied`	Shows configuration applied with the `nv config apply` command. For example, `nv show interface eth0 --applied` .
`--brief-help`	Shows help about the `nv show` command. For example, `nv show interface sw1p1 --brief-help`
`--color`	Turns colored output on or off. For example, `nv show interface eth0 --color on`
`--help`	Shows `help` for the NVUE commands.

Option	Description
`--filter`	Filters show command output on column data. For example, the `nv show interface --filter mtu=256` shows only the interfaces with operational MTU 256. To filter on multiple column outputs, enclose the filter types in parentheses; for example, `nv show interface --filter "type=ib&mtu=256"` shows data for ib interfaces with MTU 256. You can use wildcards; for example, `nv show interface swp1 --filter "link.speed=200*"` shows all interfaces that have speed start with `200` . You can filter on all revisions (operational, applied, and pending); for example, `nv show interface --filter "ip.address=1*" --rev=applied` shows all IP addresses that start with `1` for in the applied revision.
`--hostname`	Shows system configuration for the switch with the specified hostname. For example, `nv show --hostname leaf01` .
`--operational`	Shows the running configuration (the actual system state). For example, `nv show interface eth0 --operational` shows the running configuration for eth0. The running and applied configuration should be the same. If different, inspect the logs.

`--output`	Shows command output in table format (auto), `json` format or `yaml` format. For example:
	`nv show interface eth0 -o auto`
	`nv show interface eth0 -o json`
	`nv show interface eth0 -o yaml`

`--paginate`	Paginates the output. For example, `nv show interface eth0 --paginate on` .
`--pending`	Shows configuration that is `set` and `unset` but not yet applied or saved. For example, `nv show interface eth0 --pending` .
`--rev <revision>`	Shows a detached pending configuration. See the `nv config detach` configuration management command below. For example, `nv show interface eth0 --rev changeset/admin/2021-06-11_16.16.41_FPKK` .
`--startup`	Shows configuration saved with the `nv config save` command. This is the configuration after the switch boots.
`--tab`	Show information in tab format. For example, `nv show interface sw1p1 --tab.`
`--view`	Shows different views. A view is a subset of information provided by certain `nv show` commands. To see the views available for an `nv show` command, run the command with `--view` and press TAB.

4.2.9.3 Action Commands

The NVUE action commands run immediate change in the switch.

The `nv action` command is in the following categories. Each command group includes subcommands. Use command completion (Tab key) to list the subcommands.

Command Group	Description
`nv action <action> system`	Run some actions on the switch. managing the software image, reboot the system ,generate tech-support etc.
`nv action <action> interface`	Run some actions on the switch. Clear counters, renew the DHCP-client on eth0 etc.
`nv action <action> platform`	Run some actions on the platform. turn-on or turn-off the UID led etc.

4.2.9.4 Configuration Management Commands

The NVUE configuration management commands manage and apply configurations.

Command	Description
`nv config apply`	Applies the pending configuration (`nv config apply`) or a specific revision (`nv config apply 2`) to become the applied configuration. To see the list of revisions you can apply, run `nv config apply <<Tab>>` . You can also use these prompt options: • `--y` or `--assume-yes` to automatically reply `yes` to all prompts. • `--assume-no` to automatically reply `no` to all prompts. ⚠ NVOS applies but does not save the configuration; the configuration does not persist after a reboot. You can also use these apply options: `--confirm` applies the configuration change but you must confirm the applied configuration. If you do not confirm within ten minutes, the configuration rolls back automatically. You can change the default time with the apply `--confirm <time>` command. For example, `apply --confirm 60` requires you to confirm within one hour. `--confirm-status` shows the amount of time left before the automatic rollback. To save the pending configuration to the startup configuration automatically when you run `nv config apply` so that you do not have to run the `nv config save` command, enable auto save.
`nv config detach`	Detaches the configuration from the current pending configuration. NVOS names the detached configuration `pending` and includes a timestamp with extra characters. For example: `pending_20210128_212626_4WSY` To list all the current detached pending configurations, run `nv config diff <<press tab>>` .
`nv config diff <revision> <revision>`	Shows differences between configurations, such as the pending configuration and the applied configuration or the detached configuration and the pending configuration.
`nv config find <string>`	Finds a portion of the applied configuration according to the search string you provide. For example to find swp1 in the applied configuration, run `nv config find swp1` .

Command	Description
`nv config history`	Enables you to keep track of the configuration changes on the switch and shows a table with the configuration revision ID, the date and time of the change, the user account that made the change, and the type of change (such as CLI or REST API). The `nv config history <revision>` command shows the apply history for a specific revision.
`nv config patch <nvue-file>`	Updates the pending configuration with the specified YAML configuration file.
`nv config replace <nvue-file>`	Replaces the pending configuration with the specified YAML configuration file.
`nv config revision`	Shows all the configuration revisions on the switch.
`nv config save`	Overwrites the startup configuration with the applied configuration by writing to the `/etc/nvue.d/startup.yaml` file. The configuration persists after a reboot.
`nv config show`	Shows the currently applied configuration in `yaml` format. This command also shows NVUE version information.
`nv config show -o commands`	Shows the currently applied configuration commands.
`nv config diff -o commands`	Shows differences between two configuration revisions.

4.2.9.5 List All NVUE Commands

To show the full list of NVUE commands, run `nv list-commands`. For example:

```
admin@nvos:~$ nv list-commands
nv show platform environment
nv show platform environment fan
nv show platform environment fan <fan-id>
nv show platform environment temperature
nv show platform environment temperature <sensor-id>
nv show platform environment psu
nv show platform environment psu <psu-id>
nv show platform environment led
nv show platform environment led <led-id>
nv show platform software
nv show platform software installed
nv show platform software installed <installed-id>
...
```

You can show the list of commands for a command grouping. For example, to show the list of system commands:

```
admin@nvos:~$ nv list-commands system
nv show system
nv show system message
nv show system log
nv show system log file
nv show system log file <file-name>
nv show system log file list
nv show system log file follow
nv show system log component
nv show system log component <component-name>
nv show system log component <component-name> file
nv show system log component <component-name> file list
nv show system log component <component-name> file follow
nv show system log component <component-name> file <file-name>
```

```
nv show system reboot
nv show system reboot reason
...
```

Use the Tab key to get help for the command lists you want to see. For example, to show the list of command options available for the interface eth0, run:

```
admin@nvos:~$ nv list-commands interface eth0 <<press Tab>>
ip      link    pluggable
```

4.2.10 Search for a Specific Configuration

To search for a specific portion of the NVUE configuration, run the `nv config find <search string>` command. The search shows all items above and below the search string. For example, to search the entire NVUE object model configuration for any mention of `ptm`:

```
admin@nvos:~$ nv config find system
 - set:
    system:
      aaa:
        authentication:
          restrictions:
            fail-delay: 0
            lockout-state: disabled
        user:
          admin:
            password: '*'
          date-time:
            timezone: Asia/Jerusalem
```

4.2.11 Clear Switch Configuration

Users can apply an empty configuration by running the following command:

```
admin@nvos:~$ nv config apply empty
```

The above clears the current configuration, essentially returning settings to default values.

> ⚠ ACL assignments to interfaces depend on user configuration and can be removed as needed. Applying an empty configuration using `nv config apply empty` will clear all ACL assignments, rather than restoring them to their default state. To restore the default ACL assignment, run `nv unset interface acl` followed by `nv config apply`, or perform a full factory reset using `nv action reset system factory-default`.

Example Configuration Commands

This section provides examples of how to configure a NVOS switch using NVUE commands.

4.2.11.1 Configure the System Hostname

The example below shows the NVUE commands required to change the hostname for the switch to nvos:

```
admin@nvos:~$ nv set system hostname nvos
admin@nvos:~$ nv config apply
```

4.2.11.2 Configure an Interface

The example below shows the NVUE commands required to bring up sw1p1.

```
admin@nvos:~$ nv set interface sw1p1s1 link state up
admin@nvos:~$ nv config apply
```

4.2.12 Example Monitoring Commands

This section provides monitoring command examples.

4.2.12.1 Show Installed Software

The following example command lists the software installed on the switch:

```
admin@nvos:~$ nv show platform software installed
Installed software                      description
package                                 version
----------------------------------------
--------------------------------------------------------------------------------------------
----------------------------------------  ------------------------------
acl                                     access control list - utilities
acl                                     2.2.53-10
adduser                                 add and remove users and groups
adduser                                 3.118
apparmor                                user-space parser utility for AppArmor
apparmor                                2.13.6-10
apt                                     commandline package manager
apt                                     2.2.4
apt-transport-https                     transitional package for https support
apt-transport-https                     2.2.4
audisp-tacplus                          audisp module for TACACS+ accounting
audisp-tacplus                          1.0.2
auditd                                  User space tools for security auditing
auditd                                  1:3.0-2
base-files                              Debian base system miscellaneous files
base-files                              11.1+deb11u3
base-passwd                             Debian base system master password and group files
base-passwd                             3.5.51
bash                                    GNU Bourne Again SHell
bash                                    5.1-2+b3
bash-completion                         programmable completion for the bash shell
bash-completion                         1:2.11-2
bash-tacplus                            Bash TACACS+ plugin for per-command TACACS+ authorization.
bash-tacplus                            1.0.0
bridge-utils                            Utilities for configuring the Linux Ethernet bridge
bridge-utils                            1.7-1
bsdextrautils                           extra utilities from 4.4BSD-Lite
bsdextrautils                           2.36.1-8+deb11u1
bsdmainutils                            Transitional package for more utilities from FreeBSD
bsdmainutils                            12.1.7+nmu3
bsdutils                                basic utilities from 4.4BSD-Lite
bsdutils                                1:2.36.1-8+deb11u1
busybox                                 Tiny utilities for small and embedded systems
busybox                                 1:1.30.1-6+b3
ca-certificates                         Common CA certificates
ca-certificates                         20210119
...
```

4.2.12.2 Show Interface Configuration

The following example command shows the running, applied, and pending sw1p1s1 interface configuration.

```
admin@nvos::~$ nv show interface sw1p1s1
                    operational          applied
------------------  -------------------  --------
link
  auto-negotiate    on                   on
  duplex                                 full
  speed                                  auto
  counters
    in-bytes        0 Bytes
    in-pkts         0
    in-drops        0
    in-errors       0
    out-bytes       0 Bytes
    out-pkts        0
    out-drops       0
    out-errors      0
    in-symbol-errors 0
    out-wait        0
  [diagnostics]
  mtu                                    4096
  op-vls                                 VL0-VL7
  lanes                                  1X,2X,4X
  state             down                 up
  supported-speed   800G
  max-supported-mtu 4096
  physical-state    Polling
  logical-state     Down
  supported-lanes   1X,2X,4X
  vl-capabilities   VL0-VL7
 type               nvl                  nvl
```

4.2.13 Reset NVUE Configuration to Default Values

To reset the NVUE configuration on the switch back to the default values, run the following command:

```
admin@nvos:~$ nv config replace /usr/lib/python3/dist-packages/cue_config_v1/initial.yaml
admin@nvos:~$ nv config apply
```

4.2.14 Add Configuration Apply Messages

When you run the **nv config apply** command, you can add a message that describes the configuration updates you make. You can see the message when you run the **nv config history** command.

To add a configuration apply message, run the **nv config apply -m <message>** command. If the message includes more than one word, enclose the message in quotes.

```
admin@nvos:~$ nv config apply -m "this is my message"
```

4.2.15 Configure Auto Save

By default, when you run the **nv config apply** command to apply a configuration setting, NVUE applies the pending configuration to become the applied configuration and automatically saves the changes to the startup configuration file (**/etc/nvue.d/startup.yaml**).

To disable auto save so that NVUE does not save applied configuration changes, run the `nv set system config auto-save enable off` command:

```
admin@nvos:~$ nv set system config auto-save enable off
admin@nvos:~$ nv config apply
```

When you disable auto save, you must run the `nv config save` command to save the applied configuration to the startup configuration so that the changes persist after a reboot.

4.2.16 Example Configuration Management Commands

This section provides examples of how to use the configuration management commands to apply, save, and detach configurations.

4.2.16.1 Apply and Save a Configuration

The following example command configures the front panel port interfaces sw1p1 state. The configuration is only in a pending configuration state. The configuration is not applied. NVUE has not yet made any changes to the running configuration.

```
admin@nvos:~$ nv set interface sw1p1s1 link state down
```

To apply the pending configuration to the running configuration, run the `nv config apply` command. The configuration does not persist after a reboot.

```
admin@nvos:~$ nv config apply
```

To save the applied configuration to the startup configuration, run the `nv config save` command. The configuration persists after a reboot.

```
admin@nvos:~$ nv config save
```

4.2.16.2 Detach a Pending Configuration

The following example configures the IP address of the eth0 interface, then detaches the configuration from the current pending configuration. NVOS saves the detached configuration to a file with a numerical value to distinguish it from other pending configurations.

```
admin@nvos:~$ nv set interface eth0 ip address 10.10.10.1
admin@nvos:~$ nv config detach
```

4.2.16.3 View Differences Between Configurations

To view differences between configurations, run the `nv config diff` command.

To view differences between two detached pending configurations, run the `nv config diff` «TAB» command to list all the current detached pending configurations, then run the `nv config diff` command with the pending configurations you want to diff:

```
admin@nvos:~$ nv config diff <<press Tab>>
1        2        3        4        5        6        applied  empty    startup
admin@nvos:~$ nv config diff 1 2
- set:
    system:
      security:
        password-hardening:
          state: disabled
```

To view differences between the applied configuration and the startup configuration:

```
admin@nvos:~$ nv config diff applied startup
```

4.2.16.4 Replace and Patch a Pending Configuration

The following example replaces the pending configuration with the contents of the YAML configuration file called `nv-02/13/2021.yaml` located in the `/deps` directory:

```
admin@nvos:~$ nv config replace /deps/nv-02/13/2021.yaml
```

The following example patches the pending configuration (runs the set or unset commands from the configuration in the `nv-02/13/2021.yaml` file located in the `/deps` directory):

```
admin@nvos:~$ nv config patch /deps/nv-02/13/2021.yaml
```

4.2.17 Passwords and Special Characters

If you use certain special characters in a password, you must quote or escape (with a backslash) these characters so that the system understands that they are part of the password.

The following table shows if you need to quote or escape a special character.

- Normal Use indicates that you can use the special character without quotes or a backslash.
- Single Quotes and Double Quotes indicate that the entire password needs to be enclosed in quotes.

Special Character	Normal Use	Single Quotes (')	Double Quotes ("")	Escape (\)
backtick (`)	x	✓	1	✓

Special Character	Normal Use	Single Quotes (')	Double Quotes (")	Escape (\)
exclamation point (!)	x	✓	x	✓
semicolon (;)	x	✓	✓	✓
ampersand (&)	x	✓	✓	✓
question mark (?)	x	✓	✓	x
tilde (~)	x	✓	✓	✓
at-sign (@)	✓	✓	✓	✓
hash sign (#)	x	✓	✓	✓
dollar sign ($)	x	✓	x	✓
percent sign (%)	✓	✓	✓	✓
caret (^)	✓	✓	✓	✓
asterisk (*)	✓	✓	✓	✓
parentheses (())	x	✓	✓	✓
dash (-)	✓	✓	✓	✓
underscore (_)	✓	✓	✓	✓
equals sign (=)	✓	✓	✓	✓
plus sign (+)	✓	✓	✓	✓
vertical bar	x	✓	✓	✓
brackets ([])	✓	✓	✓	✓
braces ({})	✓	✓	✓	✓
colon (:)	✓	✓	✓	✓
single quote (‘)	x	x	✓	✓
double quote (“)	x	✓	x	✓
comma (,)	✓	✓	✓	✓
angle brackets (<>)	x	✓	✓	✓
slash (/)	✓	✓	✓	✓
dot (.)	2	2	2	2
white space	x	x	3	x

1. Requires escape (\) in addition to the double quotes ("").
2. You cannot use this character at the beginning of a word.
3. A word cannot consist entirely of white space, even inside double quotes.

The following example shows a password that includes a question mark (?):

```
nvos@admin:~$ nv set system aaa user nvos password “Hello?world123”
```

The following example shows a password that includes a dot (.):

```
nvos@admin:~$ nv set system aaa user nvos password “Hello.world.123”
```

The following example shows a password that includes a dot (.) and tilde (~):

```
nvos@admin:~$ nv set system aaa user nvos password “Hello.world\~123”
```

You might need to encode special characters in a password, for example in a URL. The following table shows the special character encoding.

- `✓` indicates that encoding is not needed.
- `%xx` indicates that you need to replace the special character with `%xx`.

Symbol	Normal	Single Quotes (')	Double Quotes ("")	Escape (`\`)
backtick (`)	%60	✓	%60	✓
exclamation point (`!`)	%21	✓	%21	✓
semicolon (`;`)	%3B	✓	✓	✓
ampersand (`&`)	%26	✓	✓	✓
question mark (`?`)	%3F	%3F	%3F	%3F
tilde (~)	✓	✓	✓	✓
at-sign (`@`)	✓	✓	✓	✓
hash sign (`#`)	%23	%23	%23	%23
dollar sign (`$`)	%24	✓	%24	✓
percent sign (`%`)	✓	✓	✓	✓
caret (`^`)	✓	✓	✓	✓
asterisk (`*`)	✓	✓	✓	✓
left parenthesis (`(`)	%28	✓	✓	✓
right parenthesis (`)`)	%29	✓	✓	✓
dash (`-`)	✓	✓	✓	✓
underscore (`_`)	✓	✓	✓	✓
equals sign (`=`)	✓	✓	✓	✓
plus sign (`+`)	✓	✓	✓	✓
vertical bar	%7C	✓	✓	✓
left bracket (`[`)	%5B	%5B	%5B	%5B
right bracket (`]`)	%5D	%5D	%5D	%5D

Symbol	Normal	Single Quotes (')	Double Quotes ("")	Escape (\ )
braces ({})	✓	✓	✓	✓
colon (:)	✓	✓	✓	✓
single quote (‘)	%27	%27	✓	✓
double quote (“)	%22	✓	%22	✓
comma (,)	✓	✓	✓	✓
left angle bracket (<)	%3C	✓	✓	✓
right angle bracket (>)	%3E	✓	✓	✓
slash (/)	%2F	%2F	%2F	%2F
dot (.)	✓	✓	✓	✓
white space	%20	✓	✓	✓

The following example fetches an image stored on a device with IP address 10.0.1.251 using the password `Pass#pass1` for user1:

```
nvos@admin:~$ nv action fetch system image scp://user1:Pass1%23pass1@10.0.1.251/host/nos-images/nvos-amd64-25.02.1857.bin
```

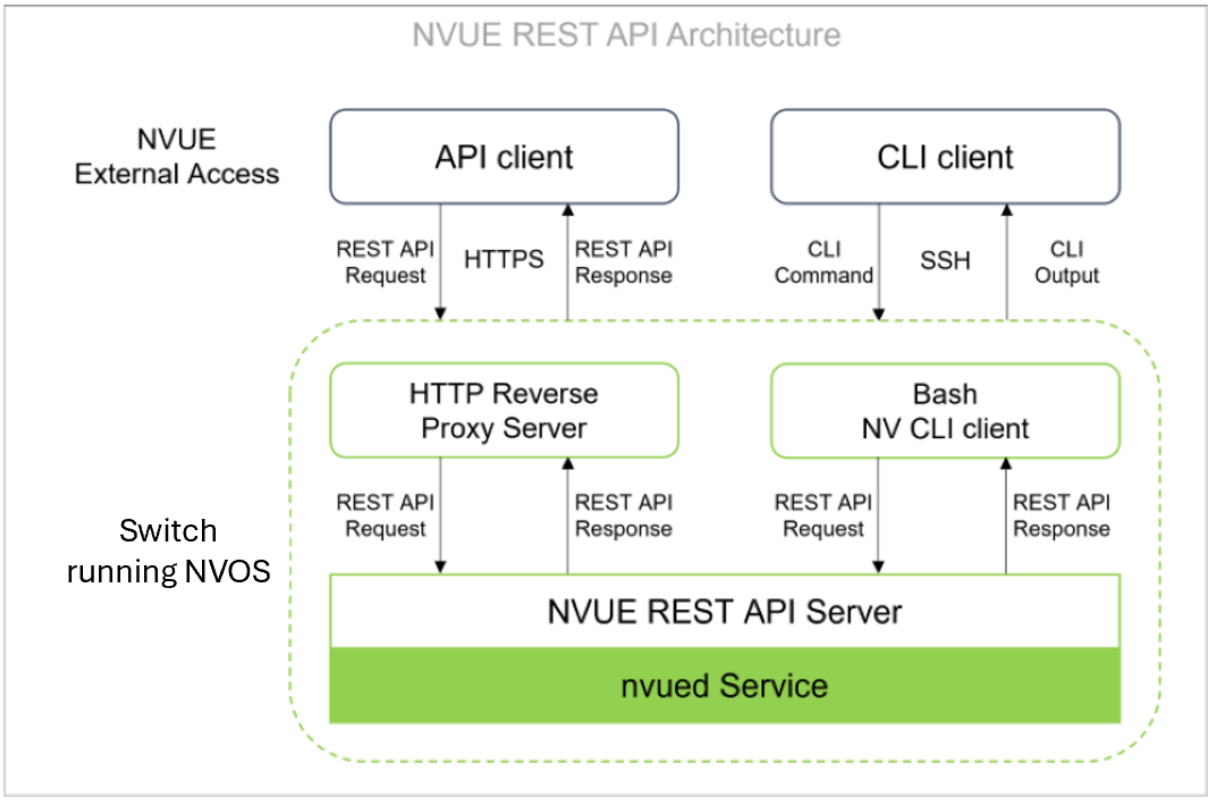
4.3 NVUE OpenAPI

This section provides information about using the NVUE API.Example Python Script

In addition to the CLI, NVUE supports a REST API. Instead of accessing NVOS using SSH, you can interact with the switch using an HTTP client, such as cURL or a web browser.

The `nued` service provides access to the NVUE REST API. NVOS exposes the HTTP endpoint internally, which makes the NVUE REST API accessible locally within the NVOS switch. The NVUE CLI also communicates with the `nvued` service using internal APIs. To provide external access to the NVUE REST API, NVOS uses an HTTP reverse proxy server, and supports HTTPS and TLS connections from external REST API clients.

The following illustration shows the NVUE REST API architecture and illustrates how NVOS forwards the requests internally.



4.3.1 Supported HTTP Methods

The NVUE REST API supports the following methods:

- The GET method displays configuration and operational data, and is equivalent to the `nv show` commands.
- The POST method creates and submits operations. You typically use this method for `nv action` commands and for the `nv config` command to create revisions.
- The PATCH method replaces or unsets a configuration. You use this method for the `nv set` and `nv config apply` commands. You can either perform:
  - A *targeted* configuration patch to make a configuration change, where you run a specific NVUE REST API targeted at a particular OpenAPI end-point URI. Based on the NVUE schema definition, you need to direct the PATCH REST API request at a particular endpoint (for example, `/nvue_v1/interface/<interface-id>/link/mtu`) and provide the payload that conforms to the schema. With a targeted configuration patch, you can control individual resources.
  - A *root* patch, where you run the NVUE PATCH API on the root node of the schema so that a single PATCH operation can change one, some, or the entire configuration in a single payload. The payload of the PATCH method must be aware of the entire NVUE object model schema because you make the configuration changes relative to the root node `/nvue_v1`. You typically perform a *root patch* to push all configurations to the switch in bulk; for example, if you use an SDN controller or a network management system to push the entire switch configuration every time you need to

make a change, regardless of how small or large. A root patch can also make configuration changes with fewer round trips to the switch.
    - The input payload in a PATCH request can have either a `set` or `unset` json object for the same resource, but not both. The order in which the API executes the `set` and `unset` objects is not deterministic and not supported.
- The DELETE method deletes a configuration and is equivalent to the `nv unset` commands.

4.3.2 Secure the API

The NVUE REST API supports HTTP basic authentication, and the same underlying authentication methods for username and password that the NVUE CLI supports. User accounts work the same on both the API and the CLI.

4.3.2.1 Certificates

NVOS includes a self-signed certificate and private key to use on the server so that it works out of the box. The switch generates the self-signed certificate and private key when it boots for the first time. The X.509 certificate with the public key is in `/etc/ssl/certs/nvue.pem` and the corresponding private key is in `/etc/ssl/private/nvue.key`.

NVIDIA recommends you use your own certificates and keys. For the steps to generate self-signed certificates and keys, refer to the Ubuntu Certificates and Security documentation.

NVOS lets you manage CA certificates (such as DigiCert or Verisign) and entity (end-point) certificates. Both a CA certificate and an entity certificate can contain a chain of certificates.

You can import certificates onto the switch (fetch certificates from an external source), set which certificate you want to use for the NVUE REST API, and show information about a certificate, such as the serial number, and the date and time during which the certificate is valid.

4.3.2.1.1 Import a Certificate

- You can import a maximum of 25 entity certificates and a maximum of 25 CA certificates.
- The certificate you import contains sensitive private key information. NVIDIA recommends that you use a secure transport such as SFTP, SCP, or HTTPS.

- To import an entity certificate, run an `nv action import system security certificate <cert-id>` command.
- To import a CA certificate, run an `nv action import system security ca-certificate <cert-id>` command.

If the certificate is passphrase protected, you need to include the passphrase.

You must provide a certificate ID (`<cert-id>`) to uniquely identify the certificate you import.

The following example imports a CA certificate with a public key and calls the certificate `tls-cert-1`. The certificate is passphrase protected with `mypassphrase`. The public key is a Base64 ASCII encoded PEM string.

```
nvos@switch:~$ nv action import system security ca-certificate tls-cert-1 passphrase mypassphrase data "<public-key>"
```

The following example imports an entity certificate bundle and calls the certificate `tls-cert-1`. The certificate bundle is passphrase protected with `mypassphrase`.

A certificate bundle must be in .PFX or .P12 format.

```
nvos@switch:~$ nv action import system security certificate tls-cert-1 passphrase mypassphrase uri-bundle scp://user@pass:1.2.3.4:/opt/certs/cert.p12
```

The following example imports an entity certificate with the public key URI `scp://user@pass:1.2.3.4` and private key URI `scp://user@pass:1.2.3.4`, and calls the certificate `tls-cert-1`. The certificate is not passphrase protected.

A CA certificate must be in .pem, .p7a, or .p7c format.

```
nvos@switch:~$ nv action import system security certificate tls-cert-1 uri-public-key scp://user@pass:1.2.3.4 uri-private-key scp://user@pass:1.2.3.4
```

4.3.2.1.2 Set the Certificate to Use

You can configure the NVUE REST API to use a specific certificate.

The following example configures the API to use the certificate `tls-cert-1`:

```
nvos@switch:~$ nv set system api certificate tls-cert-1
nvos@switch:~$ nv config apply
```

The following example configures the API to use the self-signed certificate:

```
nvos@switch:~$ nv set system api certificate self-signed
nvos@switch:~$ nv config apply
```

To unset the certificate to use with the NVUE REST API:

```
nvos@switch:~$ nv unset system api certificate tls-cert-1
```

4.3.2.1.3 Delete Certificates

- To delete an entity certificate and the key data stored on the switch, run the `nv action delete system security certificate <cert-id>` command.
- To delete a CA certificate and the key data stored on the switch, run the `nv action delete system security ca-certificate <cert-id>` command.

The following command deletes the certificate `tls-cert-1`:

```
nvos@switch:~$ nv action delete system security certificate tls-cert-1
```

4.3.2.1.4 Show Certificate Information

- To show all the entity certificates on the switch, run the `nv show system security certificate` command.
- To show all the CA certificates on the switch, run the `nv show system security ca-certificate` command.

The following example shows all the entity certificates on the switch:

```
nvos@switch:~$ nv show system security certificate
```

- To show the applications that are using a specific entity certificate, run the `nv show system security certificate <cert-id> installed` command.
- To show the applications that are using a specific CA certificate, run the `nv show system security ca-certificate <cert-id> installed` command.

The following example shows the applications that are using a specific entity certificate.

```
nvos@switch:~$ nv show system security certificate tls-cert-1 installed
```

- To show detailed information about a specific entity certificate, run the `nv show system security certificate <cert-id> dump` command.
- To show detailed information about a specific CA certificate, run the `nv show system security ca-certificate <cert-id> dump` command.

The following example shows detailed information about the CA certificate `tls-cert-1`:

```
nvos@switch:~$ nv show system security ca-certificate tls-cert-1 dump
```

4.3.2.2 Control Plane ACLs

You can secure the API by configuring:

- A listening address; see (25.03.0600-VR-0.6) NVUE OpenAPI#API Port and Listening Address below.
- Control plane ACLs; see the following example.

This example shows how to create ACLs to allow users from the management subnet and the local switch to communicate with the switch using REST APIs, and restrict all other access.

```
nvos@switch:~$ nv set acl API-PROTECT type ipv4
nvos@switch:~$ nv set acl API-PROTECT rule 10 action permit
nvos@switch:~$ nv set acl API-PROTECT rule 10 match ip .protocol tcp .dest-port 443 .source-ip 192.168.200.0/24
nvos@switch:~$ nv set acl API-PROTECT rule 10 remark "Allow the Management Subnet to talk to API"
nvos@switch:~$ nv set acl API-PROTECT rule 20 action permit
nvos@switch:~$ nv set acl API-PROTECT rule 20 match ip .protocol tcp .dest-port 443 .source-ip 127.0.0.1
nvos@switch:~$ nv set acl API-PROTECT rule 20 remark "Allow the local switch to talk to the API"
nvos@switch:~$ nv set acl API-PROTECT rule 30 action deny
nvos@switch:~$ nv set acl API-PROTECT rule 30 match ip .protocol tcp .dest-port 443
nvos@switch:~$ nv set acl API-PROTECT rule 30 remark "Block everyone else from talking to the API"
nvos@switch:~$ nv set system control-plane acl API-PROTECT inbound
```

4.3.3 Supported Objects

The NVUE object model supports most features on the NVOS switch. The following list shows the supported objects. The NVUE API supports more objects within each of these objects. You can find a full listing of the supported API endpoints here.

High-level Objects	Description
acl	Access control lists.
interface	Interface configuration.
platform	Platform configuration, such as hardware and software components.
system	Global system settings, such as system login messages, switch reboot history, syslog, ntp etc..
vrf	Get VRF.
ib	IB Devices.

4.3.4 Use the API

The NVUE CLI and the REST API are equivalent in functionality; you can run all management operations from the REST API or from the CLI. The NVUE object model drives both the REST API and the CLI management operations. All operations are consistent; for example, the CLI `nv show commands` reflect any PATCH operation (create and update) you run through the REST API.

NVUE follows a declarative model, removing context-specific commands and settings. The structure of NVUE is like a big tree that represents the entire state of a NVOS instance. At the base of the tree are high level branches representing objects, such as system and interface. Under each of these branches are more branches. As you navigate through the tree, you gain a more specific context. At the leaves of the tree are actual attributes, represented as key-value pairs. The path through the tree is similar to a filesystem path.

NVOS enables the NVUE REST API by default. To disable the NVUE REST API, run the `nv set system api state disabled` command.

> ⚠ To use the NVUE REST API in NVOS, you must change the password for the admin user; otherwise you see 401 responses when you run commands.

4.3.4.1 API Port and Listening Address

This section shows how to:

- Set the NVUE REST API port. If you do not set a port, NVOS uses the default port 443.

- Specify the NVUE REST API listening address; you can specify an IPv4 address, IPv6 address, or `localhost`. If you do not specify a listening address, NGINX listens on all addresses for the target port.

NVUE Commands

The following example sets the port to 8888:

```
nvos@switch:~$ nv set system api port 8888
nvos@switch:~$ nv config apply
```

You can listen on multiple interfaces by specifying different listening addresses:

```
nvos@switch:~$ nv set system api listening-address 10.10.10.1
nvos@switch:~$ nv set system api listening-address 10.10.20.1
nvos@switch:~$ nv config apply
```

The following example configures the listening address on eth0, which has IP address 172.0.24.0 and uses the management VRF by default:

```
nvos@switch:~$ nv set system api listening-address 172.0.24.0
nvos@switch:~$ nv config apply
```

4.3.4.1.1 Curl Command

The following example sets the port to 8888:

```
nvos@switch:~$ curl -u 'admin:****' -k --request PATCH https://localhost:443/nvue_v1/system/api?rev=2 -H 'Content-Type:application/json' -d '{"port": 8888 }'
```

You can listen on multiple interfaces by specifying different listening addresses. The following example sets localhost, interface address 10.10.10.1, and 10.10.20.1 as listen-addresses.

```
nvos@switch:~$ curl -u 'admin:****' -k --request PATCH https://localhost:443/nvue_v1/system/api/listening-address?rev=2 -H 'Content-Type:application/json' -d '{ "localhost": {}, "10.10.10.1": {}, "10.10.20.1": {}}'
```

The following example configures the listening address on eth0, which has IP address 172.0.24.0 and uses the management VRF by default:

```
nvos@switch:~$ curl -u 'admin:****' -k --request PATCH https://localhost:443/nvue_v1/system/api/listening-address?rev=2 -H 'Content-Type:application/json' -d '{"172.0.24.0": {}}'
```

4.3.4.2 Show NVUE REST API Information

To show REST API port configuration, state (enabled or disabled), certificate, listening address, and connection information:

NVUE Commands

Run the `nv show system api` command:

```
nvos@switch:~$ nv show system api
```

```
                    operational  applied
------------------  -----------  -----------
state               enabled      enabled
port                443          443
certificate         self-signed  self-signed
[listening-address] any
connections
  active            1
  accepted          2
  handled           2
  requests          2
  reading           0
  writing           1
  waiting           0
```

To show connection information only, run the **nv show system api connections** command:

```
 nvos@switch:~$ nv show system api connections
          operational
--------  -----------
active    2
accepted  8
handled   8
requests  8
reading   0
writing   2
waiting   0
```

To show the configured listening address, run the **nv show system api listening-address** command:

```
nvos@switch:~$ nv show system api listening-address
---------
localhost
```

To show all the certificates installed on the switch, run the **nv show system security certificate** command. To show information about a specific certificate, such as the serial number and how long the certificate is valid, run the **nv show system security certificate <certificate>** command:

```
 nvos@switch:~$ nv show system security certificate tls-cert-1
             operational               applied
-------------  ------------------------  -------
installed
 app           TLS
serial-number  67:03:3B:B4:6E:35:D3
valid-from     2023-02-14T00:35:18+00:00
valid-to       2033-02-11T00:35:18+00:00
```

Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -k --request GET https://localhost:443/nvue_v1/system/api?rev=2 -H "accept:
application/json"
{
  "certificate": "self-signed",
  "listening-address": {
    "10.10.10.1": {},
    "10.10.20.1": {},
    "172.0.24.0": {},
    "localhost": {}
  },
  "port": 8888,
  "state": "enabled"
}
```

To show the configured listening address:

```
nvos@switch:~$ curl -u 'admin:****' -k --request GET https://localhost:443/nvue_v1/system/api/listening-address?
rev=2 -H "accept: application/json"
{
  "10.10.10.1": {},
  "10.10.20.1": {}
}
```

To show the certificates on the switch:

```
nvos@switch:~$ curl -u 'admin:****' -k --request GET https://localhost:443/nvue_v1/system/api/certificate?rev=2 -H
"accept: application/json"
{
  "tls-cert-1": {},
  "tls-cert-2": {}
}
```

To show information about a specific certificate, such as the serial number and how long the certificate is valid:

```
nvos@switch:~$ curl -u 'admin:****' -k --request GET https://localhost:443/nvue_v1/system/api/certificate/tls-
cert-1?rev=2 -H "accept: application/json"
{
  "serial-number": "67:03:3B:B4:6E:35:D3",
  "valid-from": "2023-02-14T00:35:18+00:00",
  "valid-to": "2033-02-11T00:35:18+00:00"
}
```

4.3.4.3 Run cURL Commands

You can run the cURL commands from the command line. Use the username and password for the switch. For example:

```
nvos@switch:~$ curl  -u 'admin:****' --insecure https://127.0.0.1:443/nvue_v1/interface
{
  "eth0": {
    "ip": {
      "address": {
        "192.168.200.12/24": {}
      }
    },
    "link": {
      "mtu": 1500,
      "state": {
        "up": {}
      },
      "stats": {
        "carrier-transitions": 2,
        "in-bytes": 184151,
        "in-drops": 0,
        "in-errors": 0,
        "in-pkts": 2371,
        "out-bytes": 117506,
        "out-drops": 0,
        "out-errors": 0,
        "out-pkts": 762
      }
...
```

4.3.5 API Use Cases

The following examples show the primary API uses cases.

4.3.5.1 View a Configuration

Use the following example to obtain the current applied configuration on the switch. Change the `rev` argument to view any revision. Possible options for the `rev` argument include `startup`, `pending`, `operational`, and `applied`.

Curl Command

```
nvos@switch:~$ curl -k -u 'admin:****' -X GET "https://127.0.0.1:443/nvue_v1/?rev=applied&filled=false"
  "interface": {
    "eth0": {
      "acl": {
        "ACL_MGMT_INBOUND_CP_DEFAULT": {
          "inbound": {
            "control-plane": {}
          }
        },
        "ACL_MGMT_INBOUND_CP_DEFAULT_IPV6": {
          "inbound": {
```

```
            "control-plane": {}
          }
        },
        "ACL_MGMT_INBOUND_DEFAULT": {
          "inbound": {}
        },
        "ACL_MGMT_INBOUND_DEFAULT_IPV6": {
          "inbound": {}
        },
        "ACL_MGMT_OUTBOUND_CP_DEFAULT": {
          "outbound": {
            "control-plane": {}
          }
        },
        "ACL_MGMT_OUTBOUND_CP_DEFAULT_IPV6": {
          "outbound": {
            "control-plane": {}
          }
        }
      },
      "type": "eth"
    },
 ...
```


Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

if __name__ == "__main__":
    r = requests.get(url=nvue_end_point + "/?rev=applied&filled=false",
                     auth=auth,
                     verify=False)
    print("=======Current Applied Revision=======")
    print(json.dumps(r.json(), indent=2))
```

NVUE CLI

```
nvos@switch:~$ nv config show
- set:
    fae:
      system:
        events:
          table-size: 600
    interface:
      eth0-1:
        acl:
          ACL_MGMT_INBOUND_CP_DEFAULT:
            inbound:
              control-plane: {}
          ACL_MGMT_INBOUND_CP_DEFAULT_IPV6:
            inbound:
              control-plane: {}
          ACL_MGMT_INBOUND_DEFAULT:
            inbound: {}
          ACL_MGMT_INBOUND_DEFAULT_IPV6:
            inbound: {}
          ACL_MGMT_OUTBOUND_CP_DEFAULT:
            outbound:
              control-plane: {}
          ACL_MGMT_OUTBOUND_CP_DEFAULT_IPV6:
            outbound:
              control-plane: {}
        type: eth
...
```

4.3.5.2 Replace an Entire Configuration

To replace an entire configuration:

1. Create a new revision ID with a POST:


```
nvos@switch:~$ curl -u 'admin:****' --insecure -X POST https://127.0.0.1:443/nvue_v1/revision
{
  "1": {
    "state": "pending",
    "transition": {
      "issue": {},
      "progress": ""
    }
```

```
  }
}
```

2. Record the revision ID. In the above example, the revision ID is `"1"`.
3. Do a root patch to delete the whole configuration.

```
nvos@switch:~$ curl -u 'admin:****' -d '{}' -H 'Content-Type: application/json' -k -X DELETE https://
127.0.0.1:443/nvue_v1/?rev=1
{}
```

4. Do a root patch to update the switch with the new configuration.

```
nvos@switch:~$ curl -u 'admin:****' -d '{
    "system": {
      "hostname": "switch01"
    },
    "interface": {
      "eth0": {
        "ip": {
          "address": {
            "192.168.200.6/24": {}
          },
        },
        "type": "eth"
      },
    }
 }' -H 'Content-Type: application/json' -k -X PATCH https://127.0.0.1:443/nvue_v1/?rev=1
{}
```

5. Apply the changes with a PATCH to the revision changeset.
   Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -H 'Content-Type:application/json' -d '{"state": "apply", "auto-
prompt": {"ays": "ays_yes"}}' -k -X PATCH https://127.0.0.1:443/nvue_v1/revision/1
{
  "state": "apply",
  "transition": {
    "issue": {},
    "progress": ""
  }
}
```

   NVUE CLI

```
nvos@switch:~$ nv config apply
```

6. Review the status of the apply and the configuration:
   Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -k -X GET https://127.0.0.1:443/nvue_v1/revision/1
{
  "state": "applied",
  "transition": {
    "issue": {},
    "progress": ""
  }
}
```

```
nvos@switch:~$ curl -u 'admin:****' --insecure https://127.0.0.1:443/nvue_v1/system
{
  "contact": null,
  "date-time": {
    "local-time": "2026-05-19 08:35:26",
    "timezone": "Etc/UTC"
  },
  "health": {
    "status": "OK"
  },
  "hostname": "rosalind-eb2-164-mgmt2",
  "location": null,
  "product-name": "nvos",
  "product-release": "25.03.0514-001",
  "status": "System is ready",
```

```
  "uptime": 43910,
  "version": {
    "product-release": "25.03.0514-001"
  }
}
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def create_nvue_changest():
    r = requests.post(url=nvue_end_point + "/revision",
                      auth=auth,
                      verify=False)
    print_request(r.request)
    print_response(r)
    response = r.json()
    changeset = response.popitem()[0]
    return changeset

def apply_nvue_changeset(changeset):
    apply_payload = {"state": "apply", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/" + requests.utils.quote(changeset,
                                                               safe="")
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

def is_config_applied(changeset) -> bool:
    # Check if the configuration was indeed applied
    global RETRIES
    global POLL_APPLIED
    retries = RETRIES
    while retries > 0:
        r = requests.get(url=nvue_end_point + "/revision/" + requests.utils.quote(changeset, safe=""),
                         auth=auth,
                         verify=False)
        response = r.json()
        print(response)
        if response["state"] == "applied":
            return True
        retries -= 1
        time.sleep(POLL_APPLIED)

    return False

def apply_new_config(path,payload):
    # Create a new revision ID
    changeset = create_nvue_changest()
    print("Using NVUE Changeset: '{}'".format(changeset))

    # Delete existing configuration
    query_string = {"rev": changeset}
    r = requests.delete(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Patch the new configuration

    query_string = {"rev": changeset}
    r = requests.patch(url=nvue_end_point + path,
                       auth=auth,
                       verify=False,
                       data=json.dumps(payload),
                       params=query_string,
                       headers=mime_header)
    print_request(r.request)
    print_response(r)
```

```
    # Apply the changes to the new revision changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)

def nvue_get(path):
    r = requests.get(url=nvue_end_point + path,
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":
    payload = {
      "system": {
        "hostname": "switch01"
      },
      "interface": {
        "eth0": {
          "ip": {
            "address": {
              "192.168.200.6/24": {}
            },
          },
          "type": "eth"
        }
      },
    }
    apply_new_config("/",payload)
    time.sleep(DUMMY_SLEEP)
    print("=====Verifying some of the configurations=====")
    nvue_get("/system")
    nvue_get("/interface")
```

NVUE CLI

```
nvos@switch:~$ nv show system
                  operational             applied
----------------  ----------------------  -------
uptime            12:12:54
hostname          rosalind-eb2-164-mgmt2
product-name      nvos
product-release   25.03.0514-001
status            System is ready
date-time
  local-time      2026-05-19 08:36:30
  timezone        Etc/UTC                 Etc/UTC
health
  status          OK
version
  product-release 25.03.0514-001
```

4.3.5.3 Verify a Configuration (Dry Run)

To verify a configuration without applying it (dry run), use the dry-run option in the state-controls field when calling the apply endpoint.

The dry-run field accepts the following values:

brief: Provides basic verification output (default behavior of nv config verify)
verbose: Provides detailed output, including staged files and scheduled services

1. Create a new revision ID with a POST:

```
nvos@switch:~$ curl -u 'admin:****' --insecure -X POST https://127.0.0.1:443/nvue_v1/revision
{
 "1": {
   "state": "pending",
   "transition": {
     "issue": {},
     "progress": ""
   }
 }
}
```

2. Record the revision ID. In the above example, the revision ID is `"1"`.
3. Patch the Configuration to Verify

```
curl -u admin:admin --insecure -X PATCH -H "Content-Type: text/plain" "https://10.7.145.61:443/nvue_v1/?
rev=5" --data-binary @bad-config.txt
```

4. Run Verification (Dry Run) brief mode:

```
admin@croc-61:~$ curl -u 'admin:admin' -H 'Content-Type: application/json' -k -X PATCH -d '{"state":
"apply", "state-controls": {"dry-run": "brief"}}' --request GET https://10.7.145.61:443/nvue_v1/revision/1
```

5. Run Verification (Dry Run) Verbose mode:

```
admin@croc-61:~$ curl -u 'admin:admin' -H 'Content-Type: application/json' -k -X PATCH -d '{"state":
"apply", "state-controls": {"dry-run": "verbose"}}' --request GET https://10.7.145.61:443/nvue_v1/revision/
1
```

Verify an existing revision by id:

```
curl -u admin:admin --insecure -X PATCH -H "Content-Type: application/json" "https://10.7.145.61:443/
nvue_v1/revision/5" -d '{"state": "apply", "state-controls": {"dry-run": "brief"}}'
```

NVUE CLI

```
nvos@switch:~$ nv config verify config.txtnvos@switch:~$ nv config verify revision 5
```

4.3.5.4 Make a Configuration Change

To make a configuration change:

1. Create a new revision ID with a POST:

```
nvos@switch:~$ curl -u 'admin:****' --insecure -X POST https://127.0.0.1:443/nvue_v1/revision
{
   "2": {
   "state": "pending",
   "transition": {
     "issue": {},
     "progress": ""
   }
 }
}
```

2. Record the revision ID. In the above example, the revision ID is `"2"`.
3. Make the change with a PATCH and link it to the revision ID:
   Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -d '{"99.99.99.99/32": {}}' -H 'Content-Type: application/json' -k -X
PATCH https://127.0.0.1:443/nvue_v1/interface/eth0/ip/address?rev=2
{
  "99.99.99.99/32": {}
}
```

   NVUE CLI

```
nvos@switch:~$ nv set interface eth0 ip address 99.99.99.99/32
```

4. Apply the changes with a PATCH to the revision changeset:
   Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -H 'Content-Type:application/json' -k -X PATCH https://127.0.0.1:443/
nvue_v1/revision/2
{
  "state": "apply",
  "transition": {
    "issue": {},
    "progress": ""
  }
}
```

NVUE CLI

```
nvos@switch:~$ nv config apply
```

5. Review the status of the apply and the configuration:

Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -k -X GET https://127.0.0.1:443/nvue_v1/revision/2
{
  "state": "applied",
  "transition": {
    "issue": {},
    "progress": ""
  }
}
```

```
nvos@switch:~$ curl -u 'admin:****' -k -X GET https://127.0.0.1:443/nvue_v1/revision/2
{
  "state": "applied",
  "transition": {
    "issue": {},
    "progress": ""
  }
}
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def create_nvue_changest():
    r = requests.post(url=nvue_end_point + "/revision",
                      auth=auth,
                      verify=False)
    print_request(r.request)
    print_response(r)
    response = r.json()
    changeset = response.popitem()[0]
    return changeset

def apply_nvue_changeset(changeset):
    apply_payload = {"state": "apply", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/" + requests.utils.quote(changeset,
                                                               safe="")
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

def is_config_applied(changeset) -> bool:
```

```
    # Check if the configuration was indeed applied
    global RETRIES
    global POLL_APPLIED
    retries = RETRIES
    while retries > 0:
        r = requests.get(url=nvue_end_point + "/revision/" + requests.utils.quote(changeset, safe=""),
                         auth=auth,
                         verify=False)
        response = r.json()
        print(response)
        if response["state"] == "applied":
            return True
        retries -= 1
        time.sleep(POLL_APPLIED)

    return False

def apply_new_config(path,payload):
    # Create a new revision ID
    changeset = create_nvue_changest()
    print("Using NVUE Changeset: '{}'".format(changeset))

    # Delete existing configuration
    query_string = {"rev": changeset}
    r = requests.delete(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Patch the new configuration

    query_string = {"rev": changeset}
    r = requests.patch(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        data=json.dumps(payload),
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Apply the changes to the new revision changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)

def nvue_get(path):
    r = requests.get(url=nvue_end_point + path,
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":
    payload = {
        "99.99.99.99/32": {}
    }
    apply_new_config("/interface/eth0/ip/address",payload)
    time.sleep(DUMMY_SLEEP)
    nvue_get("/interface/eth0/ip/address")
```

NVUE CLI

```
nvos@switch:~$ nv show interface eth0 ip address

-------------
99.99.99.99/32
127.0.0.1/8
::1/128
```

4.3.6 View Differences Between Configurations

To view differences between configurations, run the API `GET /nvue_v1/<resource>?rev=<rev-A>&diff=<rev-B>` method with the configurations you want to `diff`. This method is equivalent to the NVUE `nv config diff <rev-A> <rev-B>` command.

To see the difference between the startup revision and the applied revision:

```
$ curl -u 'admin:****' --insecure -X GET /nvue_v1/interface?rev=startup&diff=applied
```

To see the difference between revision 1 and revision 2:

```
$ curl -u 'admin:****' --insecure -X GET /nvue_v1/<resource>?rev=1&diff=2
```

The order of the revisions can be changed; for example, `GET /nvue_v1/<resource>?rev=2&diff=1`.

4.3.7 Troubleshoot Configuration Changes

When a configuration change fails, you see an error in the change request.

4.3.7.1 Configuration Apply Fails with Warnings

In some cases, such as the first push with NVUE or if you change a file manually instead of using NVUE, you see a warning prompt and the apply fails.

```
nvos@switch:~$ curl -u 'admin:****' --insecure -X GET https://127.0.0.1:443/nvue_v1/revision/6
{
  "6": {
    "state": "ays_fail",
    "transition": {
      "issue": {
        "0": {
          "code": "client_timeout",
          "data": {},
          "message": "Timeout while waiting for client response",
          "severity": "error"
        }
      },
      "progress": "Aborted apply after warnings"
    }
  }
```

To resolve this issue, observe the failures or errors, then inspect the configuration that you are trying to apply. After you resolve the errors, retry the API. If you prefer to overlook the errors and force an apply, add `"auto-prompt":{"ays": "ays_yes"}` to the configuration apply.

```
nvos@switch:~$ curl -u 'admin:****' -d '{"state":"apply","auto-prompt":{"ays": "ays_yes"}}' -H 'Content-
Type:application/json' --insecure -X PATCH https://127.0.0.1:443/nvue_v1/revision/6
```

4.3.8 Save a Configuration

To save an applied configuration change to the startup configuration file (`/etc/nvue.d/startup.yaml`) so that the changes persist after a reboot, use a PATCH to the applied revision with the `save` state.

Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -k -X PATCH -d '{"state": "save", "auto-prompt": {"ays": "ays_yes"}}' -H
'Content-Type: application/json'  https://127.0.0.1:443/nvue_v1/revision/applied
{
  "state": "save",
  "transition": {
    "issue": {},
    "progress": ""
  }
}
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def save_nvue_changeset():
    apply_payload = {"state": "save", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/applied"
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":
    save_nvue_changeset()
```

NVUE CLI

```
nvos@switch:~$ nv config save
saved
```

4.3.9 Unset a Configuration Change

To unset a change, use the `null` value to the key. For example, to delete start configuration from eth1 port, use the following syntax:

```
nvos@switch:~$ curl -u 'admin:****' -d '{"eth1":null}' -H 'Content-Type: application/json' --insecure -X PATCH
https://127.0.0.1:443/nvue_v1/interface/rev=4
```

When you unset a change, you must still use the `PATCH` action. The value indicates removal of the entry. The data is `{"eth1":null}` with the PATCH action.

4.3.10 Use the API for Active Monitoring

The example below fetches the counters for `interface sw1p1` .

Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -k -X GET https://127.0.0.1:443/nvue_v1/interface/sw1p1/link/counters
{
  "buffer-overrun-errors": 0,
  "in-bytes": 16128,
  "in-drops": 0,
  "in-errors": 0,
  "in-pkts": 56,
```

```
  "in-symbol-errors": 0,
  "link-downed": 0,
  "link-error-recovery": 0,
  "local-link-integrity-errors": 0,
  "out-bytes": 16128,
  "out-drops": 0,
  "out-errors": 0,
  "out-pkts": 56,
  "out-wait": 0,
  "port-rcv-constraint-errors": 0,
  "port-rcv-remote-physical-errors": 0,
  "port-rcv-switch-relay-errors": 0,
  "qp1-drops": 0,
  "rcv-icrc-errors": 0,
  "tx-parity-errors": 0
}
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

if __name__ == "__main__":
    r = requests.get(url=nvue_end_point + "/interface/sw1p1/link/counters",
                     auth=auth,
                     verify=False)
    print("=======Interface sw1p1 Counters=======")
    print(json.dumps(r.json(), indent=2))
```

NVUE CLI

```
nvos@switch:~$ nv show interface sw1p1 link counters --view detail
                                operational  description
------------------------------  -----------
-------------------------------------------------------------------------
in-bytes                        15.75 KB     Total number of bytes received on the interface
in-pkts                         56           Total number of packets received on the interface
in-drops                        0            Number of received packets dropped
in-errors                       0            Number of received packets with errors
out-bytes                       15.75 KB     Total number of bytes transmitted out of the interface
out-pkts                        56           Total number of packets transmitted out of the interface
out-drops                       0            The number of outbound packets that were chosen to be discarded even
                                             though no errors had been detected to prevent their being

transmitted.
out-errors                      0            The number of outbound packets that could not be transmitted because
                                             of errors.
in-symbol-errors                0            Total number of minor errors detected on one or more physical lanes
out-wait                        0            The number of ticks during which the port selected by PortSelect had
                                             data to transmit but no data was sent during the entire tick because
                                             of insufficient credits or of lack of arbitration.
link-error-recovery             0            Total number of times the Port Training state machine has
successfully
                                             completed the link error recovery process
link-downed                     0            Total number of times the Port Training state machine has failed the
                                             link error recovery process and downed the link
port-rcv-remote-physical-errors 0            Total number of packets marked with the EBP delimiter received on the
                                             port
port-rcv-switch-relay-errors    0            Total number of packets received on the port that were discarded
                                             because they could not be forwarded by the switch relay
port-rcv-constraint-errors      0            Total number of packets received on the switch physical port that are
                                             discarded due to constraints
local-link-integrity-errors     0            Total number of times that the count of local physical errors
exceeded
                                             the threshold specified by Local Phy Errors
qp1-drops                       0            Total number of QP1 MADs (packets) dropped due to resource
limitations
buffer-overrun-errors           0            The number of times that Overrun-Errors consecutive flow control
                                             update periods occur
rcv-icrc-errors                 0            Total number of icrc payload corruption rx errors
tx-parity-errors                0            Total number of icrc payload corruption tx parity errors
```

4.3.11 Retrieve View Types

NVUE provides views for certain **show** commands. A view is a subset of information.

To see the views available for a show command, run the command with **--view** and press TAB:

```
nvos@switch:~$ nv show interface --view <<TAB>>
```

```
brief            detail           link-diagnostics  lldp             mac              small
```

To retrieve view types through the REST API, you use the `curl -u 'admin:****' -k -X GET http://path?view=<brief>` syntax. For example, the equivalent REST API method for the NVUE `nv show interface --view=brief` command is:

```
nvos@switch:~$ curl -u 'admin:****' -k -X GET https://127.0.0.1:443/nvue_v1/interface?view=brief
```

4.3.12 Convert CLI Changes to Use the API

You can take a configuration change from the CLI and use the API to configure the same set of changes.

1. Make your configuration changes on the system with the NVUE CLI.

```
nvos@switch:~$ nv set system hostname switch01
nvos@switch:~$ nv set interface eth0 ip address 192.168.200.6/24
```

2. View the changes as a JSON blob.

```
nvos@switch:~$ nv config diff -o json
[
  {
    "set": {
      "interface": {
        "eth0": {
          "ip": {
            "address": {
              "192.168.200.6/24": {}
            }
          }
        }
      },
      "system": {
        "hostname": "switch01"
      }
    }
  }
]
```

3. Staple the JSON blob to a root patch request as the payload.

```
nvos@switch:~$ curl -u 'admin:****' -d '{
      "interface": {
        "eth0": {
          "ip": {
            "address": {
              "192.168.200.6/24": {}
            }
          }
        }
      },
      "system": {
        "hostname": "switch01"
      }
    }' -k -X PATCH https://127.0.0.1:443/nvue_v1/?rev=3
```

4. Apply the changes with a PATCH to the revision changeset.

```
nvos@switch:~$ curl -u 'admin:****' -H 'Content-Type:application/json' -k -d '{"state": "apply", "auto-
prompt": {"ays": "ays_yes"}}' -X PATCH https://127.0.0.1:443/nvue_v1/revision/3
{
  "state": "apply",
  "transition": {
    "issue": {},
    "progress": ""
  }
}
```

5. Review the status of the apply and the configuration:
   Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -k -X GET https://127.0.0.1:443/nvue_v1/revision/3
{
  "state": "applied",
  "transition": {
    "issue": {},
    "progress": ""
  }
}
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def create_nvue_changest():
    r = requests.post(url=nvue_end_point + "/revision",
                      auth=auth,
                      verify=False)
    print_request(r.request)
    print_response(r)
    response = r.json()
    changeset = response.popitem()[0]
    return changeset

def apply_nvue_changeset(changeset):
    # apply_payload = {"state": "apply"}
    apply_payload = {"state": "apply", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/" + requests.utils.quote(changeset,
                                                               safe="")
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

def is_config_applied(changeset) -> bool:
    # Check if the configuration was indeed applied
    global RETRIES
    global POLL_APPLIED
    retries = RETRIES
    while retries > 0:
        r = requests.get(url=nvue_end_point + "/revision/" + requests.utils.quote(changeset, safe=""),
                         auth=auth,
                         verify=False)
        response = r.json()
        print(response)

        if response["state"] == "applied":
            return True
        retries -= 1
        time.sleep(POLL_APPLIED)

    return False

def apply_new_config(path,payload):
    # Create a new revision ID
    changeset = create_nvue_changest()
    print("Using NVUE Changeset: '{}'".format(changeset))

    # Delete existing configuration
    query_string = {"rev": changeset}
    r = requests.delete(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)
```

```
    # Patch the new configuration

    query_string = {"rev": changeset}
    r = requests.patch(url=nvue_end_point + path,
                       auth=auth,
                       verify=False,
                       data=json.dumps(payload),
                       params=query_string,
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Apply the changes to the new revision changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)

def nvue_get(path):
    r = requests.get(url=nvue_end_point + path,
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":
    payload = {
      "interface": {
        "eth": {
          "ip": {
            "address": {
              "192.168.200.6/24": {}
            }
          }
        }
      },
      "system": {
        "hostname": "switch01"
      }
    }
    apply_new_config("/",payload)
    time.sleep(DUMMY_SLEEP)
    nvue_get("/interface/eth0")
    nvue_get("/system")
```

4.3.13 API Examples

The following section provides practical API examples.

4.3.13.1 Configure the System

To set the system hostname, pre-login or post-login message, and time zone on the switch, send a targeted API request to `/nvue_v1/system`.

Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -d '{"system": {"hostname":"switch01","date-time": {"timezone":"America/
Los_Angeles"},"message":{"pre-login":"Welcome to NVOS","post-login":"You have successfully logged in to
switch01"}}}' -k -X PATCH https://127.0.0.1:443/nvue_v1/?rev=4
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)
```

```
def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def create_nvue_changest():
    r = requests.post(url=nvue_end_point + "/revision",
                      auth=auth,
                      verify=False)
    print_request(r.request)
    print_response(r)
    response = r.json()
    changeset = response.popitem()[0]
    return changeset

def apply_nvue_changeset(changeset):
    # apply_payload = {"state": "apply"}
    apply_payload = {"state": "apply", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/" + requests.utils.quote(changeset,
                                                               safe="")
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

def is_config_applied(changeset) -> bool:
    # Check if the configuration was indeed applied
    global RETRIES
    global POLL_APPLIED
    retries = RETRIES
    while retries > 0:
        r = requests.get(url=nvue_end_point + "/revision/" + requests.utils.quote(changeset, safe=""),
                         auth=auth,
                         verify=False)
        response = r.json()
        print(response)

        if response["state"] == "applied":
            return True
        retries -= 1
        time.sleep(POLL_APPLIED)

    return False

def apply_new_config(path,payload):
    # Create a new revision ID
    changeset = create_nvue_changest()
    print("Using NVUE Changeset: '{}'".format(changeset))

    # Delete existing configuration
    query_string = {"rev": changeset}
    r = requests.delete(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Patch the new configuration

    query_string = {"rev": changeset}
    r = requests.patch(url=nvue_end_point + path,
                       auth=auth,
                       verify=False,
                       data=json.dumps(payload),
                       params=query_string,
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Apply the changes to the new revision changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)

def nvue_get(path):
    r = requests.get(url=nvue_end_point + path,
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":
    payload = {
        "system":
        {
            "hostname":"switch01",
            "date-time":{
                "timezone":"America/Los_Angeles"
            },
            "message":
            {
                "pre-login":"Welcome to NVOS",
                "post-login:"You have successfully logged in to switch01"
            }
        }
    }
    apply_new_config("/",payload) # Root patch
    time.sleep(DUMMY_SLEEP)
```

```
    nvue_get("/system")
```

NVUE CLI

```
nvos@switch:~$ nv set system hostname switch01
nvos@switch:~$ nv set system date-time timezone America/Los_Angeles
nvos@switch:~$ nv set system message pre-login "Welcome to NVOS"
nvos@switch:~$ nv set system message post-login "You have successfully logged into switch01"
```

4.3.13.2 Configure Services

To set up NTP, SYSLOG, and SNMP on the switch, send a targeted API request to `/nvue_v1/system`.

Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -d '{"system": { "ntp": {"server":{"4.nvos.pool.ntp.org":{}}}}}' -k -X PATCH
https://127.0.0.1:443/nvue_v1/?rev=5
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def create_nvue_changest():
    r = requests.post(url=nvue_end_point + "/revision",
                      auth=auth,
                      verify=False)
    print_request(r.request)
    print_response(r)
    response = r.json()
    changeset = response.popitem()[0]
    return changeset

def apply_nvue_changeset(changeset):
    # apply_payload = {"state": "apply"}
    apply_payload = {"state": "apply", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/" + requests.utils.quote(changeset,
                                                               safe="")
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

def is_config_applied(changeset) -> bool:
    # Check if the configuration was indeed applied
    global RETRIES
    global POLL_APPLIED
    retries = RETRIES
    while retries > 0:
        r = requests.get(url=nvue_end_point + "/revision/" + requests.utils.quote(changeset, safe=""),
                         auth=auth,
                         verify=False)
        response = r.json()
        print(response)

        if response["state"] == "applied":
            return True
        retries -= 1
```

```
        time.sleep(POLL_APPLIED)

    return False

def apply_new_config(path,payload):
    # Create a new revision ID
    changeset = create_nvue_changest()
    print("Using NVUE Changeset: '{}'".format(changeset))

    # Delete existing configuration
    query_string = {"rev": changeset}
    r = requests.delete(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Patch the new configuration

    query_string = {"rev": changeset}
    r = requests.patch(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        data=json.dumps(payload),
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Apply the changes to the new revision changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)

def nvue_get(path):
    r = requests.get(url=nvue_end_point + path,
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":
    payload = {
      "system":
      {
        "ntp":
        {
          "server:
            {
              "4.nvos.pool.ntp.org": {}
            }
        },
      }
    }
    apply_new_config("/",payload) # Root patch
    time.sleep(DUMMY_SLEEP)
    nvue_get("/system/ntp")
```

NVUE CLI

```
nvos@switch:~$ nv set system ntp server 4.nvos.pool.ntp.org
```

4.3.13.3 Configure Users

The following example creates a new user, then deletes the user.

Curl Command

This example creates a new user called `test1`.

```
nvos@switch:~$ curl -u 'admin:****' -d '{"system": {"aaa": {"user": {"test1": {"hashed-
password":"72b28582708d749c6c82f3b3f226041f1bd37090281641eaeba8d44bd915d0042d609a92759d9f6fb96475cb0601cf428cd22613
df8a53a09461e0b426cf0a35","role": "monitor","state": "enabled","full-name": "Test User"}}}}}' -k -X PATCH https://
127.0.0.1:443/nvue_v1/?rev=5
```

This example deletes the `test1` user.

```
nvos@switch:~$ curl -u 'admin:****' -k -X DELETE https://127.0.0.1:443/nvue_v1/system/aaa/user/test1?rev=6
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def create_nvue_changest():
    r = requests.post(url=nvue_end_point + "/revision",
                      auth=auth,
                      verify=False)
    print_request(r.request)
    print_response(r)
    response = r.json()
    changeset = response.popitem()[0]
    return changeset

def apply_nvue_changeset(changeset):
    # apply_payload = {"state": "apply"}
    apply_payload = {"state": "apply", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/" + requests.utils.quote(changeset,
                                                               safe="")
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

def is_config_applied(changeset) -> bool:
    # Check if the configuration was indeed applied
    global RETRIES
    global POLL_APPLIED
    retries = RETRIES
    while retries > 0:
        r = requests.get(url=nvue_end_point + "/revision/" + requests.utils.quote(changeset, safe=""),
                         auth=auth,
                         verify=False)
        response = r.json()
        print(response)

        if response["state"] == "applied":
            return True
        retries -= 1
        time.sleep(POLL_APPLIED)

    return False

def apply_new_config(path,payload):
    # Create a new revision ID
    changeset = create_nvue_changest()
    print("Using NVUE Changeset: '{}'".format(changeset))

    # Delete existing configuration
    query_string = {"rev": changeset}
    r = requests.delete(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Patch the new configuration

    query_string = {"rev": changeset}
    r = requests.patch(url=nvue_end_point + path,
                       auth=auth,
                       verify=False,
                       data=json.dumps(payload),
                       params=query_string,
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Apply the changes to the new revision changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)

def delete_config(path):
    # Create an NVUE changeset
    changeset = create_nvue_changest()
```

```
    print("Using NVUE Changeset: '{}'".format(changeset))

    # Equivalent to JSON `null`
    payload = None

    # Stage the change
    query_string = {"rev": changeset}
    r = requests.delete(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        data=json.dumps(payload),
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Apply the staged changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)

def nvue_get(path):
    r = requests.get(url=nvue_end_point + path,
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":

    # Need to create a hashed password - The supported password
    # Here in this example, we use SHA-512
    import crypt
    hashed_password = crypt.crypt("hello$world#2023", salt=crypt.METHOD_SHA512)
    payload = {
        "system": {
            "aaa": {
                "user": {
                    "test1": {
                        "hashed-password": hashed_password,
                        "role": "monitor",
                        "state": "enabled",
                        "full-name": "Test User",
                    }
                }
            }
        }
    }
    apply_new_config("/",payload) # Root patch
    time.sleep(DUMMY_SLEEP)
    nvue_get("/system/user/aaa")

    """Delete an existing user account using the AAA API."""
    delete_config("/system/aaa/user/test1")
    time.sleep(DUMMY_SLEEP)
    nvue_get("/system/user/aaa")
```

NVUE CLI

This example creates a new user `test1`.

```
nvos@switch:~$ nv set system aaa user test1
nvos@switch:~$ nv set system aaa user test1 full-name "Test User"
nvos@switch:~$ nv set system aaa user test1 password "abcd@test"
nvos@switch:~$ nv set system aaa user test1 role monitor
nvos@switch:~$ nv set system aaa user test1 state enabled
```

This example deletes the user `test1`.

```
nvos@switch:~$ nv unset system aaa user test1
```

4.3.13.4 Configure an Interface

The following example configures an interface.

Curl Command

```
nvos@switch:~$ curl -u 'admin:****' -k -d '{"eth1": {"link":{"state":{"up": {}}}}}' -H 'Content-Type: application/
json' -k -X PATCH https://127.0.0.1:443/nvue_v1/interface?rev=21
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def create_nvue_changest():
    r = requests.post(url=nvue_end_point + "/revision",
                      auth=auth,
                      verify=False)
    print_request(r.request)
    print_response(r)
    response = r.json()
    changeset = response.popitem()[0]
    return changeset

def apply_nvue_changeset(changeset):
    # apply_payload = {"state": "apply"}
    apply_payload = {"state": "apply", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/" + requests.utils.quote(changeset,
                                                               safe="")
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

def is_config_applied(changeset) -> bool:
    # Check if the configuration was indeed applied
    global RETRIES
    global POLL_APPLIED
    retries = RETRIES
    while retries > 0:
        r = requests.get(url=nvue_end_point + "/revision/" + requests.utils.quote(changeset, safe=""),
                         auth=auth,
                         verify=False)
        response = r.json()
        print(response)

        if response["state"] == "applied":
            return True
        retries -= 1
        time.sleep(POLL_APPLIED)

    return False

def apply_new_config(path,payload):
    # Create a new revision ID
    changeset = create_nvue_changest()
    print("Using NVUE Changeset: '{}'".format(changeset))

    # Delete existing configuration
    query_string = {"rev": changeset}
    r = requests.delete(url=nvue_end_point + path,
                        auth=auth,
                        verify=False,
                        params=query_string,
                        headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Patch the new configuration

    query_string = {"rev": changeset}
    r = requests.patch(url=nvue_end_point + path,
                       auth=auth,
                       verify=False,
                       data=json.dumps(payload),
                       params=query_string,
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Apply the changes to the new revision changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)

def nvue_get(path):
    r = requests.get(url=nvue_end_point + path,
                     auth=auth,
```

```
                    verify=False)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":
    payload = {
      "eth1":
      {
        "type": "eth",
        "link":
        {
          "state":"up"
          }
        }
      }
    apply_new_config("/interface",payload)
    time.sleep(DUMMY_SLEEP)
    nvue_get("/interface/eth1")
```

NVUE CLI

```
nvos@switch:~$ nv set interface eth1
```

4.3.13.5 Action Operations

The NVUE action operations are ephemeral operations that do not modify the state of the configuration; they reset counters for interfaces, reboot the system etc...

Curl Command

To clear counters on sw1p1:

```
nvos@switch:~$ curl -u 'admin:****' -H 'Content-Type:application/json' -d '{"@clear": {"state": "start",
"parameters": {}}}' -k -X POST https://127.0.0.1:443/nvue_v1/interface/sw1p1/link/counters
1
nvos@switch:~$ curl -u 'admin:****' -X GET https://127.0.0.1:443/nvue_v1/action/1 -k
{"detail":"sw1p1 counters cleared.","http_status":200,"issue":[],"state":"action_success","status":"sw1p1 counters
cleared.","timeout":60,"type":""}
```

Python Code

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def nvue_action():
    r = requests.post(url=nvue_end_point + path,
                      auth=auth,
                      verify=False,
                      data=json.dumps(apply_payload),
                      headers=mime_header)
    print_request(r.request)
    print_response(r)
    return response

def nvue_get(path):
```

```
    r = requests.get(url=nvue_end_point + path,
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

if __name__ == "__main__":
    payload = {
      "@clear":
      {
        "state": "start",
        "parameters": {}
      }
    }
    action_id=nvue_action("/interface/sw1p1/link/counters",payload)
    time.sleep(DUMMY_SLEEP)
    nvue_get(f"/action/{action_id}")
```

NVUE CLI

```
nvos@switch:~$ nv action clear interface sw1p1 link counters
```

4.3.14 Example Python Script

In the following python example, the `full_config_example()` method sets the system pre-login message, and changes a few other configuration settings in a single bulk operation. The API end-point goes to the root node `/nvue_v1`.

```
#!/usr/bin/env python3

import requests
from requests.auth import HTTPBasicAuth
import json
import time

auth = HTTPBasicAuth(username="admin", password="password")
nvue_end_point = "https://127.0.0.1:443/nvue_v1"
mime_header = {"Content-Type": "application/json"}

DUMMY_SLEEP = 5  # In seconds
POLL_APPLIED = 1  # in seconds
RETRIES = 10

def print_request(r: requests.Request):
    print("=======Request=======")
    print("URL:", r.url)
    print("Headers:", r.headers)
    print("Body:", r.body)

def print_response(r: requests.Response):
    print("=======Response=======")
    print("Headers:", r.headers)
    print("Body:", json.dumps(r.json(), indent=2))

def sanity():
    # Basic retrieval to check connectivity
    r = requests.get(url=nvue_end_point + "/system",
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

def create_nvue_changest():
    r = requests.post(url=nvue_end_point + "/revision",
                      auth=auth,
                      verify=False)
    print_request(r.request)
    print_response(r)
    response = r.json()
    changeset = response.popitem()[0]
    return changeset

def apply_nvue_changeset(changeset):
    # apply_payload = {"state": "apply"}
    apply_payload = {"state": "apply", "auto-prompt": {"ays": "ays_yes"}}
    url = nvue_end_point + "/revision/" + requests.utils.quote(changeset,
                                                               safe="")
    r = requests.patch(url=url,
                       auth=auth,
                       verify=False,
                       data=json.dumps(apply_payload),
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

def full_config_example():
    # Create an NVUE changeset
    changeset = create_nvue_changest()
    print("Using NVUE Changeset: '{}'".format(changeset))
```

```
    # https://www.asciiart.eu/comics/batman
    pre_login_message = u"""
                    ,.ood888888888888boo.,
                .od888P^""             ""^Y888bo.
            .od8P''   ..oood88888888booo.     ``Y8bo.
         .odP''  .od8888888888888888888888boo.  "`Ybo.
       .d8'   od8'd888888888f`8888't888888888b`8bo   `Yb.
      d8'  od8^   8888888888[  `'  ]8888888888   ^8bo  `8b
    .8P  d88'     8888888888P      Y8888888888     `88b  Y8.
   d8' .d8'       `Y88888888'      `88888888P'       `8b. `8b
  .8P .88P            """"            """"            Y88. Y8.
  88  888                                              888  88
  88  888                                              888  88
  88  888.        ..                        ..        .888  88
  `8b `88b,     d8888b.od8bo.      .od8bo.d8888b     ,d88' d8'
   Y8. `Y88.    8888888888888b    d8888888888888    .88P' .8P
    `8b  Y88b.  `88888888888888  88888888888888'  .d88P  d8'
      Y8.  ^Y88bod8888888888888..8888888888888bod88P^  .8P
       `Y8.   ^Y8888888888888888LS8888888888888888P^   .8P'
         `^Yb.,   `^^Y88888888888888888888888P^^'   ,.dP^'
            `^Y8b..   ``^^^Y88888888P^^^'    ..d8P^'
                `^Y888bo.,            ,.od888P^'
                    "`^^Y8888888888888P^^'"
"""

    # https://www.asciiart.eu/comics/superman
    post_login_message = u'''
          _____                    _____                    _____
        //:::::::::::::::::::::::::::::::::::::::::::::::::::::\\
      //:::_______:::::::::_________:::::::::_____:::::::::\\
    //:::_/     _-"":::_--"""          """--_::::\_   )::::::::\\
   //:::/     /:::::_-"                      "-_:::\/:::::|^\:::\\
  //:::/    /~::::::I__                         \:::::::::|  \:::\\
  \\:::\    (::::::::::"""""---_____________      "--------"  /::://
   \\:::\   |:::::::::::::::::::::::::::::::"""""==_____      /::://
    \\:::"\/:::::::::::::::::::::::::::::::::::::::::::\    /~::://
     \\::::::::::::::::::::::::::::::::::::::::::::::::)/~::://
       \\::::\""""""------_____:::::::::::::::::::::::://
        \\:::"\                """""-----_____::::::://
          \\:::"\     __-----__                  )::://
            \\:::"\/~:::::::::~\_            __/~::://
              \\:::::::::::::::::""----""""::::://
                \\:::::::::::::::::::::::::::://
                  \\:::\^""--._.--""^/:::://
                    \\::"\          /"::://
                      \\::"\       /"::://
                        \\::"\_/"::://
                          \\:::::://
                            \\_//
                              "
'''

    # Prepare payload which configures a few
    # different switch configurations
    payload = {
        "interface":{
            "eth0":{
                "description": "management port"
            }
        },
        "system":{
            "message":{
                "pre-login": pre_login_message,
                "post-login": post_login_message
            },
            "date-time":{
                "timezone": "Europe/Paris"
            }
        },
    }
    # Stage the change
    query_string = {"rev": changeset}
    r = requests.patch(url=nvue_end_point + "/",  # Root patch
                       auth=auth,
                       verify=False,
                       data=json.dumps(payload),
                       params=query_string,
                       headers=mime_header)
    print_request(r.request)
    print_response(r)

    # Apply the staged changeset
    apply_nvue_changeset(changeset)

    # Check if the changeset was applied
    is_config_applied(changeset)


def message_get():
    # Get the system pre-login/post-login
    # message that was configured.
    r = requests.get(url=nvue_end_point + "/system/message",
                     auth=auth,
                     verify=False)
    print_request(r.request)
    print_response(r)

def is_config_applied(changeset) -> bool:
    # Check if the configuration was indeed applied
    global RETRIES
    global POLL_APPLIED
    retries = RETRIES
    while retries > 0:
        r = requests.get(url=nvue_end_point + "/revision/" + requests.utils.quote(changeset, safe=""),
                         auth=auth,
```

```
                        verify=False)
        response = r.json()
        print(response)

        if response["state"] == "applied":
            return True
        retries -= 1
        time.sleep(POLL_APPLIED)

    return False

if __name__ == "__main__":
    sanity()
    time.sleep(DUMMY_SLEEP)
    full_config_example()
    time.sleep(DUMMY_SLEEP)
    message_get()
```

4.3.15 Try the API

To try out the NVUE REST API, use the NVUE API Lab available on NVIDIA Air. The lab provides a basic example to help you get started. You can also try out the other examples in this document.

4.3.16 Resources

For information about using the NVUE REST API, refer to the NVUE API Swagger documentation. The full object model download is available here.

4.3.17 Considerations

- Unlike the NVUE CLI, the NVUE API does not support configuring a plain text password for a user account; you must configure a hashed password for a user account with the NVUE API.
- If you need to make multiple updates on the switch, NVIDIA recommends you use a root patch, which can make configuration changes with fewer round trips to the switch. Running many specific NVUE PATCH APIs to set or unset objects requires many round trips to the switch to set up the HTTP connection, transfer payload and responses, manage network utilization, and so on.

4.3.18 Related Information

- NGINX documentation
- Ubuntu Certificates and Security documentation
- Python requests module

4.3.19 Save the Applied Configurations

1. Send a PATCH request to save the currently applied configuration.

```
curl -u '<username:password>'--insecure --request PATCH 'https://<IP>/nvue_v1/revision/applied' -H 'Content-Type: application/json' -d '{"state": "save"}'
```

4.3.20 Send an Action

1. Run the nv action disconnect

```
$ curl -u '<username:password>'  --insecure https://<IP>/nvue_v1/system/aaa/user -H 'Content-Type: application/
json' -d '
{"@disconnect": {"state": "start", "parameters": {"user": "monitor"}
}
}'

3
```

2. Get the action ID from response; "3"

3. Check status

```
$ curl -u '<username:password>' --request GET 'https://<IP>/nvue_v1/action/3'
{"detail":"all sessions have been terminated","http_status":200,"issue":[],"state":"action_success","status":"all
sessions have been terminated","timeout":60,"type":""}
```

5 System Management

The following pages provide information on configuring general management features on the switch system.

5.1 Access Control Lists

Access Control Lists (ACLs) are rules on the switch that act as filters to manage traffic.

This section discusses the default Firewall Rules installed on the switch to protect the switch control plane and CPU from DOS and other potentially malicious network attacks and how to Configure Access Control Lists to manage traffic:

5.1.1 Firewall Rules

The NVOS default firewall rules protect the switch control plane and CPU from DOS and other potentially malicious network attacks.

The default set of firewall rules consists of IP and transport level rules. See Access Control List Configuration for custom ACL rules configurations.

Please note that users cannot bind ACL rules to the Loopback interface (lo).

5.1.1.1 DoS Rules

DoS rules protect the switch control plane and CPU from DOS attacks. NVOS provides firewall DoS rules to do the following:

- Allow only internal traffic to the loopback interfaces.
- Accept already established connections and outbound traffic.
- Drop packets if the first TCP segment is not SYN.
- Drop fragmented IP packets.
- Drop Christmas tree packets; packets with all TCP flags set.
- Drop NULL packets.
- Drop invalid packets.
- Drop strange MSS values.
- Provide brute-force protection.
- Drop packets with routing Header Type 0.
- Drop packets with a hop limit greater than 1.
- Limit excessive TCP reset packets.
- Protect against SYN flood.
- Rate limit new TCP connections for each IP address.
- Log all remaining packets, then drop them.

5.1.1.2 Whitelist Rules

Whitelist rules specify the services or application ports enabled on the switch. NVOS provides firewall whitelist rules to enable TCP ports and UDP ports.

The following table lists the ports that NVOS enables by default.

Protocol	Port	Application
TCP	22	SSH
UDP	68	DHCP Client
UDP	67	DHCP Server
UDP	123	NTP
UDP	161	SNMP
TCP	389	LDAP
TCP	636	LDAP TLS
UDP	546	DHCPv6 Client
UDP	547	DHCPv6 Server

Protocol	Port	Application
UDP	4500	IPSec-NAT
UDP	500	IKE
UDP	1812,1813,1645,1656	RADIUS
TCP	49	TACACS
UDP/TCP	53	DNS
UDP	5353	mDNS
UDP	514	remote syslog
TCP	443	HTTPS
TCP	9339	gNMI
ICMP	NA	Ping

5.1.1.3 Unset the Default Firewall Rules

The rules will be applied on first boot from the initial configuration. An unset operation will unbind them at any level. To rebind the rules, the user must explicitly use a 'set' command.

5.1.1.4 Add Firewall Rules

You cannot modify the `ACL_MGMT_INBOUND_CP_DEFAULT, ACL_MGMT_INBOUND_CP_DEFAULT_IPV6, ACL_MGMT_INBOUND_DEFAULT, ACL_MGMT_INBOUND_DEFAULT_IPV6, ACL_MGMT_OUTBOUND_CP_DEFAULT, ACL_MGMT_OUTBOUND_CP_DEFAULT_IPV6, ACL_LOOPBACK_INBOUND_CP_DEFAULT` and `ACL_LOOPBACK_INBOUND_CP_DEFAULT_IPV6` rules. However, you can append or insert additional rules.

If you use non-default ports for an application, NVIDIA recommends that you add a whitelist rule for the non-default port. For example, if you use ports 3020 and 3022 for radius server accounting and authentication instead of 1812 and 1813, you can add the following whitelist rules:

```
nvos@switch:~$ nv set acl ACL_MGMT_INBOUND_CP_DEFAULT rule 765 match ip udp source-port 3020
nvos@switch:~$ nv set acl ACL_MGMT_INBOUND_CP_DEFAULT rule 765 match ip connection-state new
nvos@switch:~$ nv set acl ACL_MGMT_INBOUND_CP_DEFAULT rule 765 match ip connection-state established
nvos@switch:~$ nv set acl ACL_MGMT_INBOUND_CP_DEFAULT rule 765 action permit
nvos@switch:~$ nv set acl ACL_MGMT_INBOUND_CP_DEFAULT rule 766 match ip udp source-port  3022
nvos@switch:~$ nv set acl ACL_MGMT_INBOUND_CP_DEFAULT rule 766 match ip connection-state new
nvos@switch:~$ nv set acl ACL_MGMT_INBOUND_CP_DEFAULT rule 766 match ip connection-state established
nvos@switch:~$ nv set acl ACL_MGMT_INBOUND_CP_DEFAULT rule 766 action permit
nvos@switch:~$ nv config apply
```

5.1.1.5 Show Firewall Rules

To show the default rules, run the `nv show acl <default-acl-id>` command, where `<default-acl-id>` is one of `ACL_MGMT_INBOUND_CP_DEFAULT, ACL_MGMT_INBOUND_CP_DEFAULT_IPV6, ACL_MGMT_INBOUND_DEFAULT, ACL_MGMT_INBOUND_DEFAULT_IPV6, ACL_MGMT_OUTBOUND_CP_DEFAULT, ACL_MGMT_OUTBOUND_CP_DEFAULT_IPV6, ACL_LOOPBACK_INBOUND_CP_DEFAULT` and `ACL_LOOPBACK_INBOUND_CP_DEFAULT_IPV6`:

```
nvos@switch:~$ nv show acl ACL_MGMT_INBOUND_CP_DEFAULT
      operational  applied
----  -----------  -------
type  ipv4         ipv4


rule
=======
    Number  Summary
    ------  ---------------------------------------
    10      action:                                 deny
            match.ip.dest-ip:            127.0.0.0/8
    20      action:                              permit
    30      action:                                deny
            match.ip.protocol:                      tcp
    40      action:                                deny
            match.ip.protocol:                      tcp
    50      action:                                deny
            match.ip.protocol:                      tcp
    60      action:                                deny
            match.ip.protocol:                      tcp
    70      action:                                deny
    80      action:                                deny
            match.ip.protocol:                      tcp
    90      action:                                deny
            match.ip.protocol:                      tcp
    100     action:                                deny
    110     match.ip.protocol:                      tcp
            match.ip.recent-list.action:            set
            match.ip.recent-list.name:              TCP
            match.ip.tcp.dest-port:                  22
    120     action:                                deny
            match.ip.protocol:                      tcp
            match.ip.recent-list.action:         update
            match.ip.recent-list.hit-count:         100
            match.ip.recent-list.name:              TCP
            match.ip.recent-list.update-interval: 60
            match.ip.tcp.dest-port:                  22
    130     match.ip.protocol:                      udp
            match.ip.recent-list.action:            set
            match.ip.recent-list.name:              UDP
            match.ip.udp.dest-port:                 161
    140     action:                                deny
            match.ip.protocol:                      udp
            match.ip.recent-list.action:         update
            match.ip.recent-list.hit-count:         100
            match.ip.recent-list.name:              UDP
            match.ip.recent-list.update-interval: 60
            match.ip.udp.dest-port:                 161
    150     match.ip.protocol:                      tcp
            match.ip.recent-list.action:            set
            match.ip.recent-list.name:              TCP
            match.ip.tcp.dest-port:                 443
    160     action:                                deny
            match.ip.protocol:                      tcp
            match.ip.recent-list.action:         update
            match.ip.recent-list.hit-count:         150
            match.ip.recent-list.name:              TCP
            match.ip.recent-list.update-interval: 60
            match.ip.tcp.dest-port:                 443
    170     match.ip.protocol:                      tcp
            match.ip.recent-list.action:            set
            match.ip.recent-list.name:              TCP
            match.ip.tcp.dest-port:                9339
    180     action:                                deny
            match.ip.protocol:                      tcp
            match.ip.recent-list.action:         update
            match.ip.recent-list.hit-count:         100
            match.ip.recent-list.name:              TCP
            match.ip.recent-list.update-interval: 60
            match.ip.tcp.dest-port:                9339
    190     match.ip.protocol:                      tcp
            match.ip.recent-list.action:            set
            match.ip.recent-list.name:              TCP
            match.ip.tcp.dest-port:                 636
    200     action:                                deny
            match.ip.protocol:                      tcp
            match.ip.recent-list.action:         update
            match.ip.recent-list.hit-count:         100
            match.ip.recent-list.name:              TCP
            match.ip.recent-list.update-interval: 60
            match.ip.tcp.dest-port:                 636
    210     match.ip.protocol:                      tcp
            match.ip.recent-list.action:            set
            match.ip.recent-list.name:              TCP
            match.ip.tcp.dest-port:                 389
    220     action:                                deny
            match.ip.protocol:                      tcp
            match.ip.recent-list.action:         update
            match.ip.recent-list.hit-count:         100
            match.ip.recent-list.name:              TCP
            match.ip.recent-list.update-interval: 60
            match.ip.tcp.dest-port:                 389
    230     match.ip.protocol:                      tcp
            match.ip.recent-list.action:            set
            match.ip.recent-list.name:              TCP
            match.ip.tcp.dest-port:                  49
    240     action:                                deny
            match.ip.protocol:                      tcp
            match.ip.recent-list.action:         update
            match.ip.recent-list.hit-count:         100
            match.ip.recent-list.name:              TCP
            match.ip.recent-list.update-interval: 60
```

```
        match.ip.tcp.dest-port:              49
250     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              123
260     action:                              deny
        match.ip.protocol:                   udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:              123
270     match.ip.protocol:                   tcp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           TCP
        match.ip.tcp.dest-port:              53
280     action:                              deny
        match.ip.protocol:                   tcp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           TCP
        match.ip.recent-list.update-interval: 60
        match.ip.tcp.dest-port:              53
290     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              53
300     action:                              deny
        match.ip.protocol:                   udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:              53
310     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              514
320     action:                              deny
        match.ip.protocol:                   udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:              514
330     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              5353
340     action:                              deny
        match.ip.protocol:                   udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:              5353
350     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              68
360     action:                              deny
        match.ip.protocol:                   udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:              68
370     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              67
380     action:                              deny
        match.ip.protocol:                   udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:              67
390     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              4500
400     action:                              deny
        match.ip.protocol:                   udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:              4500
410     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              500
420     action:                              deny
        match.ip.protocol:                   udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:              500
430     match.ip.protocol:                   udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:              1812
440     action:                              deny
```

```
        match.ip.protocol:                  udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:             1812
450     match.ip.protocol:                  udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:             1813
460     action:                            deny
        match.ip.protocol:                  udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:             1813
470     match.ip.protocol:                  udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:             1645
480     action:                            deny
        match.ip.protocol:                  udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:             1645
490     match.ip.protocol:                  udp
        match.ip.recent-list.action:         set
        match.ip.recent-list.name:           UDP
        match.ip.udp.dest-port:             1646
500     action:                            deny
        match.ip.protocol:                  udp
        match.ip.recent-list.action:      update
        match.ip.recent-list.hit-count:      100
        match.ip.recent-list.name:           UDP
        match.ip.recent-list.update-interval: 60
        match.ip.udp.dest-port:             1646
510     action:                            deny
        match.ip.hashlimit.burst:              2
        match.ip.hashlimit.expire:         30000
        match.ip.hashlimit.mode:           src-ip
        match.ip.hashlimit.name:           TCPRST
        match.ip.hashlimit.rate-above:      5/min
        match.ip.hashlimit.source-mask:        32
        match.ip.protocol:                    tcp
520     action:                            deny
        match.ip.hashlimit.burst:             30
        match.ip.hashlimit.expire:         30000
        match.ip.hashlimit.mode:           src-ip
        match.ip.hashlimit.name:        TCPGENRAL
        match.ip.hashlimit.rate-above: 50/second
        match.ip.hashlimit.source-mask:        32
        match.ip.protocol:                    tcp
530     action:                            deny
        match.ip.hashlimit.burst:             30
        match.ip.hashlimit.expire:          3000
        match.ip.hashlimit.mode:           src-ip
        match.ip.hashlimit.name:        TCPGENRAL
        match.ip.hashlimit.rate-above: 50/second
        match.ip.hashlimit.source-mask:        32
        match.ip.protocol:                    tcp
560     action:                           permit
        match.ip.protocol:                    udp
        match.ip.udp.dest-port:               161
        remark:                  Whitelist-snmp
570     action:                           permit
        match.ip.protocol:                    tcp
        match.ip.tcp.dest-port:               443
        remark:                 Whitelist-https
580     action:                           permit
        match.ip.protocol:                    tcp
        match.ip.tcp.dest-port:                22
        remark:                   Whitelist-ssh
590     action:                           permit
        match.ip.protocol:                    tcp
        match.ip.tcp.dest-port:              9339
        remark:                  Whitelist-gnmi
600     action:                           permit
        match.ip.protocol:                    tcp
        match.ip.tcp.dest-port:               636
        remark:              Whitelist-ldap-tls
610     action:                           permit
        match.ip.protocol:                    udp
        match.ip.udp.dest-port:               514
        remark:                Whitelist-rsyslog
620     action:                           permit
        match.ip.protocol:                    tcp
        match.ip.tcp.dest-port:               389
        remark:                  Whitelist-ldap
630     action:                           permit
        match.ip.protocol:                    tcp
        match.ip.tcp.dest-port:                49
        remark:                Whitelist-tacacs
640     action:                           permit
        match.ip.protocol:                    udp
        match.ip.udp.dest-port:               123
        remark:                   Whitelist-ntp
650     action:                           permit
        match.ip.protocol:                    udp
        match.ip.udp.dest-port:                53
        remark:                  Whitelist-dns
660     action:                           permit
        match.ip.protocol:                    tcp
```

```
            match.ip.tcp.dest-port:               53
            remark:                      Whitelist-dns
    670     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:               5353
            remark:                     Whitelist-mDNS
    680     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:                 68
            remark:                     Whitelist-dhcp
    690     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:                 67
            remark:                     Whitelist-dhcp
    700     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:               4500
            remark:             Whitelist-IPSec-NAT
    710     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:                500
            remark:                      Whitelist-IKE
    720     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:               1812
            remark:                   Whitelist-radius
    730     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:               1813
            remark:                   Whitelist-radius
    740     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:               1645
            remark:                   Whitelist-radius
    750     action:                             permit
            match.ip.protocol:                     udp
            match.ip.udp.dest-port:               1646
            remark:                   Whitelist-radius
    760     action:                             permit
            match.ip.protocol:                     icmp
            remark:                     Whitelist-icmp
    770     action:                                log
            match.ip.hashlimit.burst:                5
            match.ip.hashlimit.expire:    4294967295
            match.ip.hashlimit.mode:           src-ip
            match.ip.hashlimit.name:          LOGGING
            match.ip.hashlimit.rate-above:      1/min
            match.ip.hashlimit.source-mask:        32
    780     action:                              deny
```

Run the `nv show acl ACL_MGMT_INBOUND_CP_DEFAULT --rev=applied -o json` command to show additional information, such as the connection state, hit count and update interval:

```
nvos@switch:~$ nv show acl ACL_MGMT_INBOUND_CP_DEFAULT --rev=applied -o json
...
"630": {
      "action": {
        "permit": {}
      },
      "match": {
        "ip": {
          "connection-state": {
            "established": {},
            "new": {}
          },
          "protocol": "tcp",
          "tcp": {
            "dest-port": {
              "49": {}
            }
          }
        }
      },
      "remark": "Whitelist-tacacs"
    },

...

"500": {
      "action": {
        "deny": {}
      },
      "match": {
        "ip": {
          "connection-state": {
            "new": {}
          },
          "protocol": "udp",
          "recent-list": {
            "action": "update",
            "hit-count": 100,
            "name": "UDP",
            "update-interval": 60
          },
          "udp": {
            "dest-port": {
              "1646": {}
            }
```

```
          }
        }
      }
...
```

To show information about a specific rule, run the `nv show acl <default-acl-id> rule <rule>` command:

```
nvos@switch:~$ nv show acl ACL_MGMT_INBOUND_CP_DEFAULT rule 500
                      operational  applied
--------------------  -----------  -------
match
  ip
    protocol          udp          udp
    udp
      [dest-port]     1646         1646
    recent-list
      name            UDP          UDP
      update-interval 60           60
      hit-count       100          100
      action          update       update
action                deny         deny

Run the nv show acl <default-acl-id> rule <rule> --rev=applied -o json command to see additional information, such
as the connection state:


nvos@switch:~$ nv show acl ACL_MGMT_INBOUND_CP_DEFAULT rule 500 --rev=applied -o json {
  "action": {
    "deny": {}
  },
  "match": {
    "ip": {
      "connection-state": {
        "new": {}
      },
      "protocol": "udp",
      "recent-list": {
        "action": "update",
        "hit-count": 100,
        "name": "UDP",
        "update-interval": 60
      },
      "udp": {
        "dest-port": {
          "1646": {}
        }
      }
    }
  }
}
```

5.1.1.6 Log Messages

Default firewall rules include a log rule for packets that arrive in the control plane and do not match user defined or default firewall rules. The switch generates a log message in `/var/log/firewall_packet_capture.log` for packets that match the log rule.

5.1.2 Access Control List Configuration

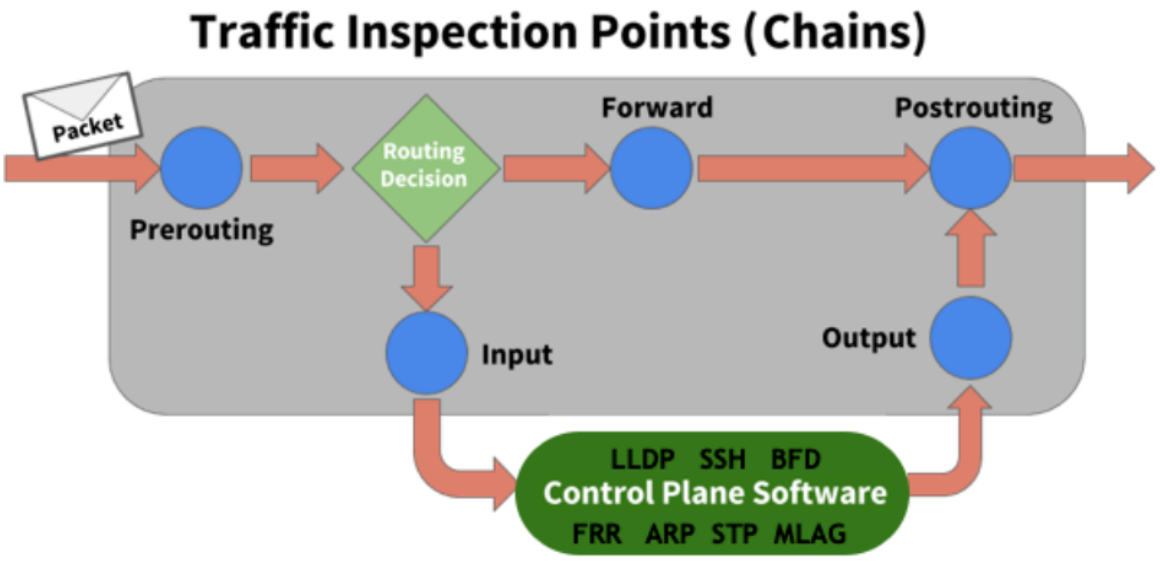
Recent-list LimiterNVOS Linux provides the NVUE, an NVOS Linux-specific userspace tool to configure custom ACLs on mgmt interfaces including 'eth0' and 'loopback' interfaces.

5.1.2.1 Traffic Rules

5.1.2.1.1 Chains

ACLs in NVOS Linux classify and control packets to and from the switch, asserting policies at layers 3 and 4 of the Open Systems Interconnection (OSI) model by inspecting packet headers according to a list of rules.

The rules inspect or operate on packets at several points (*chains*) in the life of the packet through the system:



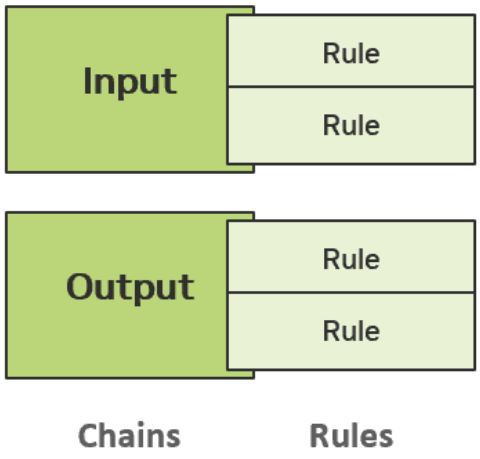
Supported chains are:

- PREROUTING touches packets before the switch routes them.

- INPUT touches packets after the switch determines that the packets are for the local system but before the control plane software receives them.
- OUTPUT touches packets from the control plane software before they leave the switch.
- POSTROUTING touches packets immediately before they leave the switch but after a routing decision.

5.1.2.1.2 Rules

Rules classify the traffic you want to control. You apply rules to chains.



5.1.2.2 Install and Manage ACL Rules with NVUE

NVOS provides a comfortable way to manage ACL rules configurations on the system using NVUE.

Consider the following example, where we want to accept packets matching the following criteria:

1. TCP packets.
2. Source IP address— `10.0.14.2/32`
3. Any source port.
4. Destination IP address— `10.0.15.8/32`
5. Any destination port.

The steps we need to perform are:

1. Set the rule type, the matching protocol, source IP address and port, destination IP address and port, and the action. You must provide a name for the rule (EXAMPLE1 in the commands below):

```
admin@nvos:~$ nv set acl EXAMPLE1 type ipv4
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip protocol tcp
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip source-ip 10.0.14.2/32
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip tcp source-port ANY
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip dest-ip 10.0.15.8/32
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip tcp dest-port ANY
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 action permit
```

2. Apply the rule to an inbound or outbound interface with the `nv set interface <interface> acl` command.
    - For rules affecting the PREROUTING or POSTROUNTING chain, apply the rule to an inbound or outbound interface accordingly: For example:

```
admin@nvos:~$ nv set interface eth0 acl EXAMPLE1 inbound
admin@nvos:~$ nv config apply
```

    - For rules affecting the INPUT or OUPUT chain, apply the rule to inbound control plane or outbound control plane interface accordingly. For example:

```
admin@nvos:~$ nv set interface eth0 acl EXAMPLE1 inbound control-plane
admin@nvos:~$ nv config apply
```

To return the default assignment of ACLs to an interface, run `nv unset interface acl` followed by `nv config apply`.

To see the configured rule run the NVUE `nv show acl <rule-name> rule <ID>` command:

```
admin@nvos:~$ nv show acl EXAMPLE1 rule 10
                    operational   applied
------------------  ------------  ------------
match
  ip
    source-ip       10.0.14.2/32  10.0.14.2/32
    dest-ip         10.0.15.8/32  10.0.15.8/32
    protocol        tcp           tcp
    tcp
      [source-port] ANY           ANY
      [dest-port]   ANY           ANY
```

To remove this rule, run the `nv unset acl <acl-name>` and `nv unset interface <interface> acl <acl-name>` commands.

```
admin@nvos:~$ nv unset acl EXAMPLE1
admin@nvos:~$ nv unset interface eth0 acl EXAMPLE1
admin@nvos:~$ nv config apply
```

To show ACL statistics per interface, such as the total number of bytes that match the ACL rule, run the `nv show interface <interface-id> acl <acl-id> statistics` or `nv show interface <interface-id> acl <acl-id> statistics <rule-id>` command, for example:

```
admin@nvos:~$ nv show interface eth0 acl EXAMPLE1 statistics
Rule  In Packet  In Byte  Out Packet  Out Byte  Layer  Remark  Action  Summary
----  ---------  -------  ----------  --------  -----  ------  ------  --------------------------------
10    0          0                              ip             permit  match.ip.dest-ip:   10.0.15.8/32
                                                                       match.ip.protocol:           tcp
                                                                       match.ip.source-ip: 10.0.14.2/32
                                                                       match.ip.tcp.dest-port:      ANY
                                                                       match.ip.tcp.source-port:    ANY
```

To see the list of all NVUE ACL commands, run the `nv list-commands acl` command.

5.1.2.2.1 Rule Support

Rule Element	Supported
Matches	Src/Dst, IP protocol In/out interface IPv4: ecn, icmp IPv6: icmpv6, routing-header, extension-header IP common: tcp (with flags), udp, multiport, frag, mss, connection-state
Standard Actions	permit, deny, log (log with log prefix), empty (no action specified)
Limiters	Recent-list Hashlimit

5.1.2.3 Default Action

The default action for each rule defined is no action. Packet matching such rules will not be affected and will continue to be processed by the proceeding rules in order.

5.1.2.4 Common Examples

5.1.2.4.1 Control Plane Limiters

NVUE allows you to configure rate limit on traffic, so incoming packets drop if they exceed certain thresholds. NVUE provides two limiters to achieve it: recent-list and hashlimit.

5.1.2.4.1.1 Recent-List Limiter

Allows you to dynamically create a list of IP addresses and then match against that list in a few different ways, for example:

```
admin@nvos:~$ nv set acl EXAMPLE2 type ipv4
admin@nvos:~$
admin@nvos:~$ nv set acl EXAMPLE2 rule 10 match ip tcp dest-port 22
admin@nvos:~$ nv set acl EXAMPLE2 rule 10 match ip protocol tcp
admin@nvos:~$ nv set acl EXAMPLE2 rule 10 match ip recent-list action set
admin@nvos:~$ nv set acl EXAMPLE2 rule 10 match ip recent-list name TCP-SSH-LIMIT
admin@nvos:~$
admin@nvos:~$ nv set acl EXAMPLE2 rule 20 match ip tcp dest-port 22
admin@nvos:~$ nv set acl EXAMPLE2 rule 20 match ip protocol tcp
admin@nvos:~$ nv set acl EXAMPLE2 rule 20 match ip recent-list action update
admin@nvos:~$ nv set acl EXAMPLE2 rule 20 match ip recent-list update-interval 60
admin@nvos:~$ nv set acl EXAMPLE2 rule 20 match ip recent-list hit-count 100
admin@nvos:~$ nv set acl EXAMPLE2 rule 20 match ip recent-list name TCP-SSH-LIMIT
admin@nvos:~$ nv set acl EXAMPLE2 rule 20 action deny
admin@nvos:~$
admin@nvos:~$ nv set interface eth0 acl EXAMPLE2 inbound control-plane
admin@nvos:~$ nv config apply
```

The above example, limits any source IP address sending more than 100 packets per 60-second interval to the switch, if it exceeds this rate all packets from this source IP address will be blocked.

Configuring the recent-list limiter consists of two consecutive rules, both rules should contain the same matching criteria (protocol tcp and dest-port 22 in the above example) and the name of the recent-list (TCP-SSH-LIMIT in the above example).

1. The first rule is set to action 'set'

2. The second rule is set to recent-list action 'update' and specified with the requested threshold using 'hit-count' and 'update-interval' (100 packets per 60-second interval in the above example)
3. The second rule action is set to deny

5.1.2.4.1.2 Hashlimit Limiter

Uses hash buckets to express a rate limiting match for a group of connections using a single rule. Grouping can be done per-hostgroup (source and/or destination address). It gives you the ability to express "*N*packets per time quantum per group", for example:

```
admin@nvos:~$ nv set acl EXAMPLE3 type ipv4
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 match ip tcp dest-port 22
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 match ip protocol tcp
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 match ip hashlimit name TCP-SSH-LIMIT
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 match ip hashlimit rate-above 5/min
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 match ip hashlimit burst 2
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 match ip hashlimit expire 3000
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 match ip hashlimit mode src-ip
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 match ip hashlimit source-mask 32
admin@nvos:~$ nv set acl EXAMPLE3 rule 10 action deny
admin@nvos:~$ nv set interface eth0 acl EXAMPLE2 inbound control-plane
admin@nvos:~$ nv config apply
```

The above example, limits any source IP address sending more than 5 packets per minute to the switch with a burst of 2 packets, if it exceeds this rate all packets from this source IP address will be blocked for 3000 milliseconds as specified in the expire parameter and will be able to send again after this period.

Configuring the recent-list limiter can be configured in one single ACL rule. The following parameters need to be configured for the hashlimit: name, rate-above, burst, expire and mode. The source-mask or destination-mask are optional.

5.1.2.4.2 Filter Specific TCP Flags

The example rule below drops ingress IPv4 TCP packets when you set the SYN bit and reset the RST, ACK, and FIN bits. The rule applies inbound on interface eth0. After configuring this rule, you cannot establish new TCP sessions that originate from ingress mgmt port eth0. You can establish TCP sessions that originate from any other port.

5.1.2.4.2.1 NVUE Commands

```
admin@nvos:~$ nv set acl EXAMPLE4 type ipv4
admin@nvos:~$ nv set acl EXAMPLE4 rule 20 match ip protocol tcp
admin@nvos:~$ nv set acl EXAMPLE4 rule 20 match ip tcp flags syn
admin@nvos:~$ nv set acl EXAMPLE4 rule 20 match ip tcp mask rst
admin@nvos:~$ nv set acl EXAMPLE4 rule 20 match ip tcp mask syn
admin@nvos:~$ nv set acl EXAMPLE4 rule 20 match ip tcp mask fin
admin@nvos:~$ nv set acl EXAMPLE4 rule 20 match ip tcp mask ack
admin@nvos:~$ nv set acl EXAMPLE4 rule 20 action deny
admin@nvos:~$ nv set interface eth0 acl EXAMPLE4 inbound
admin@nvos:~$ nv config apply
```

5.1.2.4.3 Control Who Can SSH into the Switch

Run the following commands to control who can SSH into the switch. In the following example, 10.10.10.1/32 is the interface IP address (or loopback IP address) of the switch and 10.255.4.0/24 can SSH into the switch.

5.1.2.4.3.1 NVUE Commands

```
admin@nvos:~$ nv set acl example2 type ipv4
admin@nvos:~$ nv set acl example2 rule 10 match ip source-ip 10.255.4.0/24
admin@nvos:~$ nv set acl example2 rule 10 match ip dest-ip 10.10.10.1/32
admin@nvos:~$ nv set acl example2 rule 10 action permit
admin@nvos:~$ nv set acl example2 rule 20 match ip source-ip ANY
admin@nvos:~$ nv set acl example2 rule 20 match ip dest-ip 10.10.10.1/32
admin@nvos:~$ nv set acl example2 rule 20 action deny
admin@nvos:~$ nv set interface eth0 acl example2 inbound
admin@nvos:~$ nv config apply
```

5.1.2.4.4 Match on ECN Bits in the TCP IP Header

ECN allows end-to-end notification of network congestion without dropping packets. You can add ECN rules to match on the ECE, CWR, and ECT flags in the TCP IPv4 header.

By default, ECN rules match a packet with the bit set. You can reverse the match by using an explanation point (!).

5.1.2.4.4.1 Match on the ECE Bit

After an endpoint receives a packet with the CE bit set by a router, it sets the ECE bit in the returning ACK packet to notify the other endpoint that it needs to slow down.

To match on the ECE bit:

NVUE Commands

```
admin@nvos:~$ nv set acl example2 type ipv4
admin@nvos:~$ nv set acl example2 rule 10 match ip protocol tcp
admin@nvos:~$ nv set acl example2 rule 10 match ip ecn flags tcp-ece
admin@nvos:~$ nv set acl example2 rule 10 action permit
admin@nvos:~$ nv set interface eth0 acl example2 inbound
admin@nvos:~$ nv config apply
```

5.1.2.4.4.2 Match on the CWR Bit

The CWR bit notifies the other endpoint of the connection that it received and reacted to an ECE.

To match on the CWR bit:

NVUE Commands

```
admin@nvos:~$ nv set acl example2 type ipv4
admin@nvos:~$ nv set acl example2 rule 10 match ip protocol tcp
admin@nvos:~$ nv set acl example2 rule 10 match ip ecn flags tcp-cwr
admin@nvos:~$ nv set acl example2 rule 10 action permit
admin@nvos:~$ nv set interface eth0 acl example2 inbound
admin@nvos:~$ nv config apply
```

5.1.2.4.4.3 Match on the ECT Bit

The ECT codepoints negotiate if the connection is ECN capable by setting one of the two bits to 1. Routers also use the ECT bit to indicate that they are experiencing congestion by setting both the ECT codepoints to 1.

To match on the ECT bit:

NVUE Commands

```
admin@nvos:~$ nv set acl example2 type ipv4
admin@nvos:~$ nv set acl example2 rule 10 match ip protocol tcp
admin@nvos:~$ nv set acl example2 rule 10 match ip ecn ip-ect 1
admin@nvos:~$ nv set acl example2 rule 10 action permit
admin@nvos:~$ nv set interface eth0 acl example2 inbound
admin@nvos:~$ nv config apply
```

5.1.2.4.5 Set DSCP on Transit Traffic

The following examples use the *mangle* table to modify the packet as it transits the switch. DSCP is in decimal notation in the examples below.

To set SSH as high priority traffic:

```
admin@nvos:~$ nv set acl EXAMPLE1 type ipv4
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip protocol tcp
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip tcp dest-port 22
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 action set dscp 46
admin@nvos:~$ nv set interface eth0 acl EXAMPLE1 inbound
admin@nvos:~$ nv config apply
```

To set everything coming in swp1 as AF13:

```
admin@nvos:~$ nv set acl EXAMPLE1 type ipv4
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 action set dscp 14
admin@nvos:~$ nv set interface eth0 acl EXAMPLE1 inbound
admin@nvos:~$ nv config apply
```

To set Packets destined for 10.0.100.27 as best effort:

```
admin@nvos:~$ nv set acl EXAMPLE1 type ipv4
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip dest-ip 10.0.100.27/32
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 action set dscp 0
admin@nvos:~$ nv set interface eth0 acl EXAMPLE1 inbound
admin@nvos:~$ nv config apply
```

To use a range of ports for TCP traffic:

```
admin@nvos:~$ nv set acl EXAMPLE1 type ipv4
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip protocol tcp
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip source-ip 10.0.0.17/32
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip tcp source-port 10000:20000
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip dest-ip 10.0.100.27/32
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 match ip tcp dest-port 10000:20000
admin@nvos:~$ nv set acl EXAMPLE1 rule 10 action set dscp 34
admin@nvos:~$ nv set interface eth0 acl EXAMPLE1 inbound
admin@nvos:~$ nv config apply
```

To specify all ports on the switch in NVUE (swp+ in an iptables rule), you must set the range of interfaces on the switch as in the examples above (**nv set interface swp1-48**). This command creates as many rules in the **/etc/cumulus/acl/policy.d/50_nvue.rules** file as the number of interfaces in the range you specify.

5.1.3 Access Control List Commands

5.1.3.1 nv show acl

	nv show acl Display all available ACLs on the system.
Syntax Description	N/A
History	25.02.2002 25.02.70xx Updated output
Example	admin@nvos:~$ nv show acl Type Summary --- acl-default-dos ipv4 rule: 10 rule: 20 rule: 30 acl-default-dos-ipv6 ipv6 rule: 10 rule: 20 rule: 30 acl-default-loopback ipv4 rule: 10 acl-default-loopback-ipv6 ipv6 rule: 10 acl-default-outbound ipv4 rule: 10 rule: 20 acl-default-outbound-ipv6 ipv6 rule: 10 rule: 20 acl-default-whitelist ipv4 rule: 10 rule: 20 acl-default-whitelist-ipv6 ipv6 rule: 10 rule: 20 rule: 30
REST API	GET https://<ip>/nvue_v1/acl
Related Commands	nv set acl
Notes	

5.1.3.2 nv unset acl

	nv unset acl Clear all the new configured ACLs and restore the original default ACLs.
Syntax Description	N/A
History	25.02.2002
Example	admin@nvos:~$ nv unset acl
REST API	DELETE https://<ip>/nvue_v1/acl
Related Commands	nv show acl
Notes	This command will remove the modifications/extra ACLs configured on the system and restore to the original default ACLs.

5.1.3.3 nv show acl id

	nv show acl <acl-id> Get ACL <acl-id> information (i.e., rule-ids and the ACL type: ipv4 or ipv6).	
Syntax Description	acl-id	ACL name
History	25.02.2002	
Example	admin@nvos:~$ nv show acl acl_1 operational applied ---- ----------- ------- type ipv4 ipv4 rule ======= Number Summary ------ ---------- 1 action: permit match.ip.protocol: tcp	
REST API	GET https://<ip>/nvue_v1/acl/<acl-id>	
Related Commands	nv show acl	
Notes		

5.1.3.4 nv set/unset acl id

	nv set acl <acl-id> nv unset acl <acl-id> Create a new custom ACL Delete an existing ACL.	
Syntax Description	acl-id	New, custom ACL name
History	25.02.2002	
Example	admin@nvos:~$nv set acl EXAMPLE_ACL	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/<acl-id>	
Related Commands	nv show acl	
Notes	• ACL name can be chosen to any generic name but is important later on binding multiple ACLs on the same interface and same direction since ACLs list of rules will be ordered with lexicographical order. • For example, ACL with name 'A' that has 10 rules and acl with name 'B' with 5 rules, if bound to the same direction on the same interface, the 10 rules of acl 'A' will be before the 5 rules of acl 'B'. • This command is not enough for applying this custom acl, it needs to have at **least one rule** in it and needs to belong to the ip **type either ipv4 or ipv6.** • The unset command will not remove the specified ACL if it is bound to an interface. The user must unbind it and then use this command to delete the ACL. • Unset of default ACL will restore the original list of rules of that ACL.	

5.1.3.5 nv set/unset acl type

	nv set acl <acl-id> type <acl-type> nv unset acl <acl-id> type <acl-type> Add ACL type, whether it is an IPv4 or IPv6 ACL.	
Syntax Description	acl-id	New, custom ACL name
	acl-type	Enum: ipv4 | ipv6
History	25.02.2002	
Example		
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/<acl-id>/type/	
Related Commands	nv show acl	
Notes	Each ACL must have a type	

5.1.3.6 nv show acl rule

	nv show acl <acl-id> rule Display all the rules configured on the specified ACL.	
Syntax Description	acl-id	ACL name
History	25.02.2002	
Example		
REST API	GET https://<ip>/nvue_v1/acl/<acl-id>/rule	
Related Commands	nv show acl <acl-id>	
Notes		

```
admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule
Number  Summary
------  ---------------------------------
10      action:                    permit
        match.ip.protocol:            udp
        match.ip.udp.dest-port:       161
        remark:             Whitelist-snmp
20      action:                    permit
        match.ip.protocol:            tcp
        match.ip.tcp.dest-port:       443
        remark:            Whitelist-https
30      action:                    permit
        match.ip.protocol:            tcp
        match.ip.tcp.dest-port:        22
        remark:             Whitelist-ssh
```

5.1.3.7 nv show acl rule id

	nv show acl <acl-id> rule <rule-id> Show ACL rule <rule-id> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)

History	25.02.2002
Example	admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 10 operational applied remark Whitelist-snmp Whitelist-snmp match ip protocol udp udp udp [dest-port] 161 161 [connection-state] new new [connection-state] established established action permit permit
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	

5.1.3.8 nv set/unset acl rule

	nv set acl <acl-id> rule <rule-id> nv unset acl <acl-id> rule <rule-id> Set/remove ACL rule <rule-id> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl user_custom_acl rule 10	
REST API	PATH https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• This command is used to declare the specified rule with the specified ACL. • Mere application of configuration is insufficient. Matching criteria on either the packet or action for this rule must be specified in order to be effective.	

5.1.3.9 nv set/unset acl rule remark

	nv set acl <acl-id> rule <rule-id> remark <string> nv unset acl <acl-id> rule <rule-id> remark <string> Set/remove ACL rule <rule-id> remark configurations (remark is the same as description).	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl user_custom_acl rule 20 remark "MY-PROTECTIVE-RULE"	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/action/deny	

Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	The remark acts the same as a description of a rule.

5.1.3.10 nv show acl rule action

	nv show acl <acl-id> rule <rule-id> action Show ACL rule <rule-id> action configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 10 action operational applied ----------- ------- permit permit	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/action	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.11 nv set/unset acl rule action permit

	nv set acl <acl-id> rule <rule-id> action permit nv unset acl <acl-id> rule <rule-id> action permit Set/remove ACL rule <rule-id> action permit.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl user_custom_acl rule 10 action permit	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/action/permit	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Only one action per rule can be specified. • Any rule matching the specified rule will be accepted to the system. • Leaving a rule with no action will make the action as "permit" by default	

5.1.3.12 nv set/unset acl rule action deny

	nv set acl <acl-id> rule <rule-id> action deny nv unset acl <acl-id> rule <rule-id> action deny Set/remove ACL rule <rule-id> action deny.

Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl user_custom_acl rule 20 action deny`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/action/deny	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Only one action per rule can be specified. • Any rule matching the specified rule will be rejected by the system and will be processed any further. • Leaving a rule with no action will leave any packet matching the specified rule unaffected.	

5.1.3.13 nv set/unset acl rule action log log-prefix

	nv set acl <acl-id> rule <rule-id> action log log-prefix <str> nv unset acl <acl-id> rule <rule-id> action log log-prefix <str> Set/remove ACL rule <rule-id> action log log-prefix <str>.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	log-prefix-str	String
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl user_custom_acl rule 20 action log` `admin@nvos:~$ nv set acl user_custom_acl rule 30 action log log-prefix "Dropped-by-custom-acl"`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/action/log PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/action/log/log-prefix/<log-prefix-str>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Only one action per rule can be specified. • Log-prefix can be set to empty string. • Any packet matching specified rule with logging action will be logged to netfilter log. • Leaving a rule with no action will leave any packet matching the specified rule unaffected.	

5.1.3.14 nv show acl rule match

	nv show acl <acl-id> rule <rule-id> match Show ACL rule <rule-id> match configuration.

Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 10 match
	                    operational applied
	------------------ ----------- -----------
	ip
	  protocol udp udp
	  udp
	    [dest-port] 161 161
	  [connection-state] new new
	  [connection-state] established established</pre>	

REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	Currently, displaying the matching criteria for the rule only contains layer 3 and 4 filtering criteria in the OSI model.	

5.1.3.15 nv set/unset acl rule match

	nv set acl <acl-id> rule <rule-id> match nv unset acl <acl-id> rule <rule-id> match Set/remove ACL rule <rule-id> match.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	<pre>admin@nvos:~$ nv set acl user_custom_acl rule 20 match</pre>	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Leaving a rule with empty matching criteria will cause the rule to match any packet. • The unset form of the command will remove the match criteria of the rule.	

5.1.3.16 nv show acl rule match ip

	nv show acl <acl-id> rule <rule-id> match ip Show ACL rule <rule-id> match IP configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	

Example	admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 10 match ip operational applied protocol udp udp udp [dest-port] 161 161 [connection-state] new new [connection-state] established established
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	Displays the matching IP criteria for the rule.

5.1.3.17 nv set/unset acl rule match ip

	nv set acl <acl-id> rule <rule-id> match ip nv unset acl <acl-id> rule <rule-id> match ip Set/remove ACL rule <rule-id> match ip configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv unset acl user_custom_acl rule 20 match ip	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Leaving a rule with empty IP matching criteria will cause the rule to match any packet. • The unset command will remove the IP match criteria of the rule.	

5.1.3.18 nv show acl rule match ip udp

	nv show acl <acl-id> rule <rule-id> match ip udp Show ACL rule <rule-id> match IP UDP configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 10 match ip udp operational applied [dest-port] 161 161	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/udp	
Related Commands	nv set acl <acl-id> rule <rule-id>	

Notes	• Display the matching UDP IP criteria for the rule. • The output primarily contains either the source port or destination port.

5.1.3.19 nv show acl rule match ip udp dest-port

	nv show acl <acl-id> rule <rule-id> match ip udp dest-port Show ACL rule <rule-id> match IP UDP dest-port configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 10 match ip udp dest-port Ports ----- 161	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/udp/dest-port	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• The command displays the matching dest-port of UDP IP criteria for the rule. • The rule can have more than dest-port configured.	

5.1.3.20 nv set/unset acl rule match ip udp dest-port

	nv set acl <acl-id> rule <rule-id> match ip udp dest-port <port-num> nv unset acl <acl-id> rule <rule-id> match ip udp dest-port <port-num> Configure/remove ACL rule <rule-id> match IP UDP dest-port <port-num> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	port-num	IP port ID (integer: 0–65535 | enum: ANY, bootpc, bootps, clag, dhcp-client, dhcp-server, domain, ftp, http, https, imap2, ldap, ldaps, ntp, msdp, pop3, smtp, snmp, snmp-trap, ssh, telnet, tftp | ip-port-range)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl custom-acl rule 650 match ip udp dest-port 22 admin@nvos:~$ nv set acl custom-acl rule 650 match ip udp dest-port 53	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/udp/dest-port/<port-num>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• The rule can have more than dest-port configured. • Rule cannot be configured with more than one source port and more than one dest-port at the same time in the same rule. For example, the user cannot configure ports 22, 53 on the dest-port and 1813 on the source-port, but can configure 22 on dest-port and 1813 on source-port. • The user can configure more than one port on dest-port or source-port.	

5.1.3.21 nv show acl rule match ip udp source-port

	nv show acl <acl-id> rule <rule-id> match ip udp source-port Show ACL rule <rule-id> match IP UDP source-port configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	 admin@nvos:~$ nv show acl custom-acl rule 650 match ip udp source-port Ports ----- 53 22	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/udp/source-port	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Display the matching dest-port of UDP IP criteria for the rule. • The rule can have more than dest-port configured.	

5.1.3.22 nv set/unset acl rule match ip udp source-port

	nv set acl <acl-id> rule <rule-id> match ip udp source-port <port-num> nv unset acl <acl-id> rule <rule-id> match ip udp source-port <port-num> Configure/remove ACL rule <rule-id> match IP UDP source-port <port-num> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	port-num	IP port ID (integer: 0–65535 | enum:ANY, bootpc, bootps, clag, dhcp-client, dhcp-server, domain, ftp,http, https, imap2, ldap, ldaps, ntp, msdp, pop3, smtp,snmp, snmp-trap,ssh, telnet, tftp | ip-port-range)
History	25.02.2002	
Example	 admin@nvos:~$ nv set acl custom-acl rule 650 match ip udp source-port 22 admin@nvos:~$ nv set acl custom-acl rule 650 match ip udp source-port 53	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/udp/source-port/<port-num>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• The rule can have more than source-port configured. • Rule cannot be configured with more than one source port and more than one dest-port at the same time in the same rule. For example, the user cannot configure ports 22, 53 on the dest-port and 1813 on the source-port, but can configure 22 on dest-port and 1813 on source-port. • The user can configure more than one port on dest-port or source-port.	

5.1.3.23 nv show acl rule match ip tcp

	nv show acl <acl-id> rule <rule-id> match ip tcp Show ACL rule <rule-id> match ip tcp configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 230 match ip tcp operational applied ---------- ---------- ------- [dest-port] 9351 9351 [dest-port] 9352 9352 [dest-port] 9353 9353 [dest-port] 9370 9370 admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 50 match ip tcp operational applied ---------- ---------- ------- [dest-port] 9339 9339	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Display the matching UDP IP criteria for the rule. • The output primarily contains any of the source-port, dest-port, flags, mask, mss, all-mss-except.	

5.1.3.24 nv show acl rule match ip tcp dest-port

	nv show acl <acl-id> rule <rule-id> match ip tcp dest-port Show ACL rule <rule-id> match IP TCP dest-port configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv show acl acl-default-whitelist-ipv6 rule 230 match ip tcp dest-port Ports ----- 9351 9352 9353 9370	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/dest-port	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Display the matching dest-port of TCP IP criteria for the rule. • The rule can have more than dest-port configured.	

5.1.3.25 nv set/unset acl rule match ip tcp dest-port

	nv set acl <acl-id> rule <rule-id> match ip tcp dest-port <port-num> nv unset acl <acl-id> rule <rule-id> match ip tcp dest-port <port-num> Configure/remove ACL rule <rule-id> match ip tcp dest-port <port-num> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	port-num	IP port ID (integer: 0–65535 | enum: ANY, bootpc, bootps, clag, dhcp-client, dhcp-server, domain, ftp,http, https, imap2, ldap, ldaps, ntp, msdp, pop3, smtp,snmp, snmp-trap,ssh, telnet, tftp | ip-port-range)
History	25.02.2002	
Example	 admin@nvos:~$ nv set acl custom-acl rule 650 match ip tcp dest-port 22 admin@nvos:~$ nv set acl custom-acl rule 650 match ip tcp dest-port 53	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/dest-port/<port-num>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• The rule can have more than source-port configured. • Rule cannot be configured with more than one source port and more than one dest-port at the same time in the same rule. For example, the user cannot configure ports 22, 53 on the dest-port and 1813 on the source-port, but can configure 22 on dest-port and 1813 on source-port. • The user can configure more than one port on dest-port or source-port.	

5.1.3.26 nv show acl rule match ip tcp source-port

	nv show acl <acl-id> rule <rule-id> match ip tcp source-port Show ACL rule <rule-id> match IP TCP source-port configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	 admin@nvos:~$ nv show acl custom-acl rule 650 match ip tcp source-port Ports ----- 53 22	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/source-port	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Display the matching dest-port of UDP IP criteria for the rule. • The rule can have more than the dest-port configured.	

5.1.3.27 nv set/unset acl rule match ip tcp source-port

	nv set acl <acl-id> rule <rule-id> match ip tcp source-port <port-num> nv unset acl <acl-id> rule <rule-id> match ip tcp source-port <port-num> Configure/remove ACL rule <rule-id> match ip tcp source-port <port-num> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	port-num	IP port ID (integer: 0–65535 | enum:ANY, bootpc, bootps, clag, dhcp-client, dhcp-server, domain, ftp,http, https, imap2, ldap, ldaps, ntp, msdp, pop3, smtp,snmp, snmp-trap,ssh, telnet, tftp | ip-port-range)
History	25.02.2002	
Example	 admin@nvos:~$ nv set acl custom-acl rule 650 match ip tcp source-port 22 admin@nvos:~$ nv set acl custom-acl rule 650 match ip tcp source-port 53	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/source-port/<port-num>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• The rule can have more than source-port configured. • Rule cannot be configured with more than one source port and more than one dest-port at the same time in the same rule. For example, the user cannot configure ports 22, 53 on the dest-port and 1813 on the source-port, but can configure 22 on dest-port and 1813 on source-port. • The user can configure more than one port on dest-port or source-port.	

5.1.3.28 nv show acl rule match ip tcp flags

	nv show acl <acl-id> rule <rule-id> match ip tcp flags Show ACL rule <rule-id> match ip tcp flags configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	

Example	
	admin@nvos:~$ nv show acl ACL_MGMT_INBOUND_CP_DEFAULT rule 60 match ip tcp flags -o json
	{
	"none": {}
	}	

REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/flags	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.29 nv set/unset acl rule match ip tcp flags

	nv set acl <acl-id> rule <rule-id> match ip tcp flags (syn | ack | fin | rst | urg | psh | all | none) nv unset acl <acl-id> rule <rule-id> match ip tcp flags (syn | ack | fin | rst | urg | psh | all | none) Configure/remove ACL rule <rule-id> match ip tcp flags <flag-id> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	flag-id	enum: (syn | ack | fin | rst | urg | psh | all | none)
History	25.02.2002	
Example	 admin@nvos:~$ nv set acl custom-acl rule 650 match ip tcp flags all admin@nvos:~$ nv set acl custom-acl rule 660 match ip tcp flags urg admin@nvos:~$ nv set acl custom-acl rule 660 match ip tcp flags psh admin@nvos:~$ nv set acl custom-acl rule 660 match ip tcp flags syn	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/flags/<flag-id>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• The user can configure multiple flags that are not 'none' or 'all'. • The flag configurations must come with TCP mask configurations.	

5.1.3.30 nv show acl rule match ip tcp mask

	nv show acl <acl-id> rule <rule-id> match ip tcp mask ACL rule <rule-id> match IP TCP mask configuration.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	

Example	
	admin@nvos:~$ nv show acl ACL_MGMT_INBOUND_CP_DEFAULT rule 60 match ip tcp mask -o json
	{
	"ack": {},
	"fin": {},
	"rst": {},
	"syn": {}
	}	

REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/mask	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.31 nv set/unset acl rule match ip tcp mask

	nv set acl <acl-id> rule <rule-id> match ip tcp mask (syn | ack | fin | rst | urg | psh | all | none) nv unset acl <acl-id> rule <rule-id> match ip tcp mask (syn | ack | fin | rst | urg | psh | all | none) Configure/remove ACL rule <rule-id> match ip tcp mask <flag-id> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	flag-id	enum: (syn | ack | fin | rst | urg | psh | all | none)
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv set acl custom-acl rule 650 match ip tcp mask all
	admin@nvos:~$ nv set acl custom-acl rule 660 match ip tcp mask urg
	admin@nvos:~$ nv set acl custom-acl rule 660 match ip tcp mask psh
	admin@nvos:~$ nv set acl custom-acl rule 660 match ip tcp mask syn</pre>	

REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/flags/<flag-id>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• The user can configure multiple flags that are not 'none' or 'all'. • The flag configurations must come with TCP mask configurations.	

5.1.3.32 nv set/unset acl rule match ip tcp mss

	nv set acl <acl-id> rule <rule-id> match ip tcp mss <mss-format> nv unset acl <acl-id> rule <rule-id> match ip tcp mss <mss-format> Configure/remove ACL rule <rule-id> match ip tcp mss configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	mss-format	tcpmss value could be an integer or a range. Examples: "0-1", "536-65535", "65000", "128"
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv set acl custom-acl rule 650 match ip tcp mss 536
	admin@nvos:~$ nv set acl custom-acl rule 660 match ip tcp mss 536-65535</pre>	

REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/mss/<mss-format>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	The command will match TCP packets with the specified MSS values.	

5.1.3.33 nv set/unset acl rule match ip tcp all-mss-except

	nv set acl <acl-id> rule <rule-id> match ip tcp all-mss-except <mss-format> nv unset acl <acl-id> rule <rule-id> match ip tcp all-mss-except <mss-format> Configure/remove ACL rule <rule-id> match ip tcp all-mss-except configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	mss-format	tcpmss value could be an integer or a range. Examples: "0-1", "536-65535", "65000", "128"
History	25.02.2002	
Example	 admin@nvos:~$ nv set acl custom-acl rule 650 match ip tcp all-mss-except 536 admin@nvos:~$ nv set acl custom-acl rule 660 match ip tcp all-mss-except 536-65535	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/tcp/all-mss-except/<mss-format>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	The command will match all TCP packets with MSS value different than the specified MSS values.	

5.1.3.34 nv set/unset acl rule match ip fragment

	nv set acl <acl-id> rule <rule-id> match ip fragment nv unset acl <acl-id> rule <rule-id> match ip fragment Configure/remove ACL rule <rule-id> match IP fragment configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	 admin@nvos:~$ nv set acl custom-acl rule 650 match ip fragment	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/fragment	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	Match fragmented packets.	

5.1.3.35 nv show acl rule match ip ecn

	nv show acl <acl-id> rule <rule-id> match ip ecn Configure/remove ACL rule <rule-id> match IP ECN configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)

History	25.02.2002
Example	admin@nvos:~$nv show acl b rule 1 match ip ecn operational applied ip-ect 3 3
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/ecn
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	

5.1.3.36 nv set/unset acl rule match ip ecn

	nv set acl <acl-id> rule <rule-id> match ip ecn nv unset acl <acl-id> rule <rule-id> match ip ecn Configure/remove ACL rule <rule-id> match IP ECN configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv unset acl custom-acl rule 650 match ip ecn	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/ecn	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.37 nv set/unset acl rule match ip ecn ip-ect

	nv set acl <acl-id> rule <rule-id> match ip ecn ip-ect <ip-ect-num> nv unset acl <acl-id> rule <rule-id> match ip ecn ip-ect <ip-ect-num> Configure/remove ACL rule <rule-id> match IP ECN ip-ect configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	ip-ect	ip-ect (integer: 0-3)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl custom-acl rule 650 match ip ecn ip-ect 0	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/ecn/ip-ect	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.38 nv set/unset acl rule match ip ecn flags

	nv set acl <acl-id> rule <rule-id> match ip ecn flags <ecn-flag> nv unset acl <acl-id> rule <rule-id> match ip ecn flags <ecn-flag> Configure/remove ACL rule <rule-id> match IP ECN ip-ect configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	en-flag	enum: tcp-cwr | tcp-ece
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 650 match ip ecn flags  tcp-cwr`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/ecn/flags/<flag-id>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.39 nv show acl rule match ip connection-state

	nv show acl <acl-id> rule <rule-id> match ip connection-state Show ACL rule <rule-id> match IP connection-state configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	

Example	`admin@nvos:~$  nv show acl acl-default-whitelist-ipv6 rule 10 match ip connection-state`
	`operational  applied`
	`-----------  -----------`
	`new          new`
	`established  established`	

REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/connection-state	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.40 nv set/unset acl rule match ip connection-state

	nv set acl <acl-id> rule <rule-id> match ip connection-state <state-id> nv unset acl <acl-id> rule <rule-id> match ip connection-state <state-id> Configure/remove ACL rule <rule-id> match IP connection-state <state-id> configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	state-id	state-id can be: established, invalid, new, related

History	25.02.2002
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip connection-state new`
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/ecn/ip-ect
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	Multiple connection-states can be configured.

5.1.3.41 nv show acl rule match ip extension-header

	nv show acl <acl-id> rule <rule-id> match ip extension-header Show ACL rule <rule-id> match IP extension-header configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	

Example	`admin@nvos:~$ nv show acl ACL1 rule 1 match ip extension-header`
	`operational applied`
	`---------- ---------- ----------`
	`type hop-by-hop hop-by-hop`	

REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/extension-header	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.42 nv set/unset acl rule match ip extension-header type

	nv set acl <acl-id> rule <rule-id> match ip extension-header type <hop-by-hop> nv unset acl <acl-id> rule <rule-id> match ip extension-header type <hop-by-hop> Configure/remove ACL rule <rule-id> match IP extension-header configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip extension-header type hop-by-hop`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/extension-header/type/<type>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• This configuration is relevant to IPv6 ACLs only. • Matches '-m hbh' in ip6tables tool	

5.1.3.43 nv show acl rule match ip routing-header

	nv show acl <acl-id> rule <rule-id> match ip routing-header Show ACL rule <rule-id> match ip routing-header configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv show acl ACL_MGMT_INBOUND_CP_DEFAULT_IPV6 rule 850 match ip routing-header operational applied ---- ----------- ------- type 0 0	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/routing-header	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.44 nv set/unset acl ACL rule match ip routing-header type

	nv set acl <acl-id> rule <rule-id> match ip routing-header type <hop-by-hop> nv unset acl <acl-id> rule <rule-id> match ip routing-header type <hop-by-hop> Configure/remove ACL rule <rule-id> match ip routing-header configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl custom-acl rule 10 match ip extension-header type 0	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/routing-header/type/<type>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• This configuration is relevant to IPv6 ACLs only. • Matches '-m rt' in ip6tables tool	

5.1.3.45 nv set/unset acl ACL rule match ip source-ip

	nv set acl <acl-id> rule <rule-id> match ip source-ip <ip-format> nv unset acl <acl-id> rule <rule-id> match ip source-ip <ip-format> Configure/remove ACL rule <rule-id> match ip source-ip configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	source-ip	(ANY | <ipv4> | <ipv6> | <ipv4-prefix> | <ipv6-prefix> | <ipv4-netmask> | <ipv6-netmask>)

History	25.02.2002
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip source-ip 127.0.0.1/8`
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/source-ip/<ip-format>
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	The user cannot configure IPv4 address on an ACL with IPv6 and vice versa.

5.1.3.46 nv set/unset acl ACL rule match ip dest-ip

	nv set acl <acl-id> rule <rule-id> match ip dest-ip <ip-format> nv unset acl <acl-id> rule <rule-id> match ip dest-ip <ip-format> Configure/remove ACL rule <rule-id> match ip dest-ip configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	dest-ip	(ANY | <ipv4> | <ipv6> | <ipv4-prefix> | <ipv6-prefix> | <ipv4-netmask> | <ipv6-netmask>)
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip dest-ip 127.0.0.1/8`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/dest-ip/<ip-format>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	The user cannot configure IPv4 address on an ACL with IPv6 and vice versa.	

5.1.3.47 nv set/unset acl rule match ip protocol

	nv set acl <acl-id> rule <rule-id> match ip protocol <protocol-format> nv unset acl <acl-id> rule <rule-id> match ip protocol <protocol-format> Configure/remove ACL rule <rule-id> match IP dest-ip configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	protocol-format	(0-255 | tcp | udp | icmp | icmpv6)
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip protocol tcp`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/protocol/<protocol-format>	

Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	

5.1.3.48 nv set/unset acl rule match ip icmp-type

	nv set acl <acl-id> rule <rule-id> match ip icmp-type <icmp-format> nv unset acl <acl-id> rule <rule-id> match ip icmp-type <icmp-format> Configure/remove ACL rule <rule-id> match IP ICMP-type configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	icmp-format	(0-255 | echo-reply | echo-request | time-exceeded | dest-unreachable | port-unreachable)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl custom-acl rule 10 match ip icmp-type echo-reply admin@nvos:~$ nv set acl custom-acl rule 10 match ip icmp-type 9	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/icmp-type/<icmp-format>	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• The protocol must be specified to be ICMP $nv set acl custom-acl rule 10 match ip protocol icmp • IPv4 type must be specified for the configured ACL.	

5.1.3.49 nv set/unset acl rule match ip icmpv6-type

	nv set acl <acl-id> rule <rule-id> match ip icmpv6-type <icmp-format> nv unset acl <acl-id> rule <rule-id> match ip icmpv6-type <icmp-format> Configure/remove ACL rule <rule-id> match IP ICMPv6-type configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	icmpv6-format	(0-255 | router-solicitation | router-advertisement | neighbor-solicitation | neighbor-advertisement)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl custom-acl rule 10 match ip icmp-type router-solicitation admin@nvos:~$ nv set acl custom-acl rule 10 match ip icmp-type 9	

REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/icmpv6-type/<icmpv6-format>
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	• The protocol must be specified to be ICMPv6. `$nv set acl custom-acl rule 10 match ip protocol icmpv6` • IPv6 type must be specified for the configured ACL.

5.1.3.50 nv show acl rule match ip recent-list

	nv show acl <acl-id> rule <rule-id> match ip recent-list Show ACL rule <rule-id> match ip recent-list configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	

Example	`admin@nvos:~$ nv show acl acl-default-dos rule 120 match ip recent-list`
	`operational applied`
	`--- --- ---`
	`name TCP TCP`
	`update-interval 60 60`
	`hit-count 100 100`
	`action update update`	

REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/recent-list	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Matches the 'recent' iptables module. • Used to filter IP address that passes a specific rate. In the above example, the rate is 100 packets per 60 seconds, if a source-ip sends more than this rate, the IP address will be blocked.	

5.1.3.51 nv set/unset acl rule match ip recent-list name

	nv set acl <acl-id> rule <rule-id> match ip recent-list name <generic-name> nv unset acl <acl-id> rule <rule-id> match ip recent-list name <generic-name> Configure/remove ACL rule <rule-id> match IP recent-list name configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip recent-list name "EXAMPLE"`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/recent-list	
Related Commands	nv set acl <acl-id> rule <rule-id>	

Notes	• In order to configure recent-list, the user needs to configure action and name for the recent-list
	  • For action set: configure name and action=set
	  • For action update: configure name and action=update, hit-count and update-interval
	• There can be multiple recent-lists in the system and each is distinguished by a name.
	• Refer to the documentation of 'recent' in 'iptables' for further information.

5.1.3.52 nv set/unset acl rule match ip recent-list action

	nv set acl <acl-id> rule <rule-id> match ip recent-list action (set | update) nv unset acl <acl-id> rule <rule-id> match ip recent-list action (set | update) Configure/remove ACL rule <rule-id> match IP recent-list action configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip recent-list action update`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/recent-list	
Related Commands	nv set acl <acl-id> rule <rule-id>	

Notes	• In order to configure recent-list, the user needs to configure action and name for the recent-list
	  • For action set: configure name and action=set
	  • For action update: configure name and action=update, hit-count and update-interval
	• There can be multiple recent-lists in the system and each is distinguished by a name.
	• Refer to the documentation of 'recent' in 'iptables' for further information.	

5.1.3.53 nv set/unset acl rule match ip recent-list hit-count

	nv set acl <acl-id> rule <rule-id> match ip recent-list hit-count (1-4294967295) nv unset acl <acl-id> rule <rule-id> match ip recent-list hit-count (1-4294967295) Configure/remove ACL rule <rule-id> match ip recent-list hit-count configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip recent-list  hit-count  100`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/recent-list	
Related Commands	nv set acl <acl-id> rule <rule-id>	

Notes	• In order to configure recent-list, the user needs to configure action and name for the recent-list
	  • For action set: configure name and action=set
	  • For action update: configure name and action=update, hit-count and update-interval
	• There can be multiple recent-lists in the system and each is distinguished by a name.
	• Refer to the documentation of 'recent' in 'iptables' for further information.

5.1.3.54 nv set/unset acl rule match ip recent-list update-interval

	nv set acl <acl-id> rule <rule-id> match ip recent-list update-interval (1-4294967295) nv unset acl <acl-id> rule <rule-id> match ip recent-list update-interval (1-4294967295) Configure/remove ACL rule <rule-id> match ip recent-list update-interval configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	 admin@nvos:~$ nv set acl custom-acl rule 10 match ip recent-list update-interval 60	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/recent-list	
Related Commands	nv set acl <acl-id> rule <rule-id>	

Notes	• In order to configure recent-list, the user needs to configure action and name for the recent-list
	  • For action set: configure name and action=set
	  • For action update: configure name and action=update, hit-count and update-interval
	• There can be multiple recent-lists in the system and each is distinguished by a name.
	• Refer to the documentation of 'recent' in 'iptables' for further information.	

5.1.3.55 nv show acl rule match ip hashlimit

	nv show acl <acl-id> rule <rule-id> match ip hashlimit Show ACL rule <rule-id> match ip hashlimit configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	 admin@nvos:~$ nnv show acl acl-default-dos rule 600 match ip hashlimit operational applied ---------- ---------- ---------- name LOGGING LOGGING rate-above 1/min 1/min burst 5 5 source-mask 32 32 expire 4294967295 4294967295 mode src-ip src-ip	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/recent-list	

Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	• Matches the 'hashlimit' iptables module. • This is another way to filter IP addresses. • The following properties are required for configuration: name, rate-above, burst, expire, mode.

5.1.3.56 nv set/unset acl rule match ip hashlimit name

	nv set acl <acl-id> rule <rule-id> match ip hashlimit name <generic-name> nv unset acl <acl-id> rule <rule-id> match ip hashlimit name <generic-name> Configure/remove ACL rule <rule-id> match IP hashlimit name configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip hashlimit name "Limiter"`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/hashlimit	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Matches the 'hashlimit' iptables module. • The following properties are required for configuration: name, rate-above, burst, expire, mode.	

5.1.3.57 nv set/unset acl rule match ip hashlimit rate-above

	nv [un]set acl <acl-id> rule <rule-id> match ip hashlimit rate-above <rate-format> Configure/remove ACL rule <rule-id> match IP hashlimit rate configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	rate-format	Rate limit, should be in the following format: integer/time-unit where time-unit is one of [second | min | hour]. The max supported rate is 1000000/second
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip hashlimit rate 2/min`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/hashlimit	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Matches the 'hashlimit' iptables module. • The following properties are required for configuration: name, rate-above, burst, expire, mode.	

5.1.3.58 nv set/unset acl rule match ip hashlimit burst

	nv [un]set acl <acl-id> rule <rule-id> match ip hashlimit burst <burst-int> Configure/remove ACL rule <rule-id> match IP hashlimit burst configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	burst-int	integer:1–4294967295
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip hashlimit burst 5`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/hashlimit	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Matches the 'hashlimit' iptables module. • The following properties are required for configuration: name, rate-above, burst, expire, mode.	

5.1.3.59 nv set/unset acl rule match ip hashlimit expire

	nv [un]set acl <acl-id> rule <rule-id> match ip hashlimit expire <expire-int> Configure/remove ACL rule <rule-id> match IP hashlimit expire configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	expire-int	integer:1–4294967295
History	25.02.2002	
Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip hashlimit expire 3`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/hashlimit	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Matches the 'hashlimit' iptables module. • The following properties are required for configuration: name, rate-above, burst, expire, mode.	

5.1.3.60 nv set/unset acl rule match ip hashlimit mode

	nv set acl <acl-id> rule <rule-id> match ip hashlimit mode <mode> nv unset acl <acl-id> rule <rule-id> match ip hashlimit mode <mode> Configure/remove ACL rule <rule-id> match IP hashlimit mode configurations.

Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	mode	(enum:src-ip, dst-ip | string)
History	25.02.2002	
Example	admin@nvos:~$ nv set acl custom-acl rule 10 match ip hashlimit mode src-ip	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/hashlimit	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Matches the 'hashlimit' iptables module. • The following properties are required for configuration: name, rate-above, burst, expire, mode.	

5.1.3.61 nv set/unset acl rule match ip hashlimit destination-mask

	nv set acl <acl-id> rule <rule-id> match ip hashlimit destination-mask <mask> nv unset acl <acl-id> rule <rule-id> match ip hashlimit destination-mask <mask> Configure/remove ACL rule <rule-id> match IP hashlimit destination-mask configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	mask	integer: for ipv4 the range is 0-32 and for ipv6 the range is 0-128
History	25.02.2002	
Example	admin@nvos:~$ nv set acl custom-acl rule 10 match ip hashlimit destination-mask 32	
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/hashlimit	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	• Matches the 'hashlimit' iptables module. • The following properties are required for configuration: name, rate-above, burst, expire, mode.	

5.1.3.62 nv set/unset acl rule match ip hashlimit source-mask

	nv set acl <acl-id> rule <rule-id> match ip hashlimit source-mask <mask> nv unset acl <acl-id> rule <rule-id> match ip hashlimit source-mask <mask> Configure/remove ACL rule <rule-id> match ip hashlimit source-mask configurations.	
Syntax Description	acl-id	ACL name
	rule-id	Rule number (integer: 1–65535)
	mask	Integer: IPv4 range: 0–32 IPv6 range: 0–128
History	25.02.2002	

Example	`admin@nvos:~$ nv set acl custom-acl rule 10 match ip hashlimit source-mask 32`
REST API	PATCH/DELETE https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/match/ip/hashlimit
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	• Matches the 'hashlimit' iptables module. • The following properties are required for configuration: name, rate-above, burst, expire, mode.

5.1.3.63 nv show interface acl

	nv show interface <iface-id> acl Display the ACL bound to the interface.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002 25.02.70xx Updated output	
Example		
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

```
admin@nvos:~$ nv show interface eth0 acl
ACL Name                  Rule ID  In Packets  In Bytes  Out Packets  Out Bytes
------------------------  -------  ----------  -------   -----------  ---------
acl-default-dos           10       0           0 Bytes
                          20       260         20.45 KB
                          30       0           0 Bytes
                          40       0           0 Bytes
                          50       0           0 Bytes
acl-default-dos-ipv6      10       0           0 Bytes
                          20       0           0 Bytes
                          30       0           0 Bytes
                          40       0           0 Bytes
acl-default-outbound      10                               0            0 Bytes
                          20                               295          32.96 KB
acl-default-outbound-ipv6 10                               0            0 Bytes
                          20                               0            0 Bytes
acl-default-whitelist     10       0           0 Bytes
                          20       0           0 Bytes
                          30       1083        82.80 KB
acl-default-whitelist-ipv6 10      0           0 Bytes
                          20       0           0 Bytes
                          30       0           0 Bytes
```

5.1.3.64 nv show interface lo acl

	nv show interface lo acl Display the acl bound to the loopback interface.
Syntax Description	N/A
History	25.02.2002 25.02.70xx Updated output

Example	admin@nvos:~$ nv show interface lo acl ACL Name Rule ID In Packets In Bytes Out Packets Out Bytes ---------- --------- ---------- ---------- ---------- ---------- acl-default-loopback 10 0 0 Bytes acl-default-loopback-ipv6 10 0 0 Bytes
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	

5.1.3.65 nv show interface acl id

	nv show interface <iface-id> acl <acl-id> Display the given acl-id bound to the interface.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002 25.02.70xx Updated output	
Example	admin@nvos:~$ nv show interface eth0 acl acl-default-whitelist Statistics ========== Rule In Packet In Byte Out Packet Out Byte Action Match ---- ---------- --------- ---------- --------- ---------- ---------- 10 0 0 Bytes permit ip udp dest- port : 161 connection-state new : False established : False protocol : udp	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.66 nv show interface acl statistics

	nv show interface <iface-id> acl <acl-id> statistics Display the given acl-id statistics bound to the interface.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002 25.02.70xx Updated output	

Example	admin@nvos:~$ nv show interface eth0 acl acl-default-whitelist statistics Rule In Packet In Byte Out Packet Out Byte Action Match ---- --------- -------- --------- --------- ------------------ ---------- 10 0 0 Bytes permit ip udp dest- port : 161 connection- state new : False established : False protocol : udp
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/statistics
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	

5.1.3.67 nv show interface acl statistics id

	nv show interface <iface-id> acl <acl-id> statistics <rule-id> Display the given acl-id statistics bound to the interface.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002 25.02.70xx Updated output	
Example	admin@nvos:~$ nv show interface eth0 acl acl-default-whitelist statistics 10 operational applied ---------- ----------- ------- match ip protocol udp udp [dest-port] 161 [connection-state] new [connection-state] established action permit inbound packet 0 byte 0 Bytes	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/statistics/{rule-id}	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes		

5.1.3.68 nv show interface acl outbound

	nv show interface <iface-id> acl <acl-id> outbound Display the given acl-id bound to the interface in the outbound direction.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002 25.02.70xx Updated output	

Example	admin@nvos:~$ nv show interface eth0 acl acl-default-whitelist outbound Statistics ========== Rule Out Packet Out Byte Action Match ---- ---------- ---------- ---------- ---------- 10 0 0 Bytes permit ip udp dest-port : 161 connection-state new : False established : False protocol : udp 20 0 0 Bytes permit ip tcp dest-port : 443 connection-state new : False established : False protocol : tcp
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/outbound
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	If an ACL is configured on one direction and not the other, it will be shown in the parent show (nv show interface <iface> acl <acl-id>) and not in the show of the direction it is not configured on.

5.1.3.69 nv show interface acl outbound control-plane

	nv show interface <iface-id> acl <acl-id> outbound control-plane Display the given acl-id bound to the interface in the outbound control-plane direction.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002 25.02.70xx Updated output	
Example	admin@nvos:~$ nv show interface eth0 acl acl-default-outbound outbound control-plane Statistics ========== Rule Out Packet Out Byte Action Match ---- --------- -------- ---------- ---------- 10 0 0 Bytes deny ip connection- state invalid : False 20 47 4824 Bytes permit	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/outbound/control-plane	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	If an ACL is configured on one direction and not the other, it will be shown in the parent show (nv show interface <iface> acl <acl-id>) and not in the show of the direction it is not configured on.	

5.1.3.70 nv show interface acl inbound

	nv show interface <iface-id> acl <acl-id> inbound Display the given acl-id bound to the interface in the inbound direction.

Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002 25.02.70xx Updated output	
Example	admin@nvos:~$ nv show interface eth0 acl acl-default-whitelist inbound Statistics ========== Rule In Packet In Byte Action Match ---- --------- -------- ---------- 10 0 0 Bytes permit ip udp dest-port : 161 connection-state new : False established : False protocol : udp	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/inbound	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	If an ACL is configured on one direction and not the other, it will be shown in the parent show (nv show interface <iface> acl <acl-id>) and not in the show of the direction it is not configured on.	

5.1.3.71 nv show interface acl inbound control-plane

	nv show interface <iface-id> acl <acl-id> Display the given acl-id bound to the control-plane in the inbound direction.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002 25.02.70xx Updated output	
Example	admin@nvos:~$ nv show interface eth0 acl acl-default-whitelist inbound control-plane Statistics ========== Rule In Packet In Byte Action Match ---- --------- -------- ---------- 10 0 0 Bytes permit ip udp dest-port : 161 connection-state new : False established : False protocol : udp	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/inbound/control-plane	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	If an ACL is configured on one direction and not the other, it will be shown in the parent show (nv show interface <iface> acl <acl-id>) and not in the show of the direction it is not configured on.	

5.1.3.72 nv set/unset interface acl inbound

	nv set interface <iface-id> acl <acl-id> inbound nv unset interface <iface-id> acl <acl-id> inbound Configure/remove the binding of the given ACL on the specified interface.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
	acl-id	ACL name
History	25.02.2002	
Example	`admin@nvos:~$ nv set interface eth0 acl ACL1 inbound`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/inboun	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	An ACL cannot be bound to inbound and inbound control-plane or cannot be bound to outbound and outbound control-plane on the same interface!	

5.1.3.73 nv set/unset interface acl inbound control-plane

	nv set interface <iface-id> acl <acl-id> inbound control-plane nv unset interface <iface-id> acl <acl-id> inbound control-plane Configure the binding of the given ACL on the specified interface. The unset form of the command removes the binding of the given ACL on the specified interface.	
Syntax Description	iface-id	interface could be one of 'eth0' or 'loopback'
	acl-id	ACL name
History	25.02.2002 25.02.70xx Updated syntax	
Example	`admin@nvos:~$ nv set interface eth0 acl acl-default-dos inbound control-plane` `admin@nvos:~$ nv unset interface eth0 acl acl-default-whitelist inbound control-plane`	
REST API	PATCH https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/inbound/control-plane DELETE https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/inbound/control-plane	
	nv set acl <acl-id> rule <rule-id>	
Notes	An ACL cannot be applied to both inbound and inbound control-plane, or to both outbound and outbound control-plane, on the same interface. Resetting the firewall rules to their default settings on the interface restores normal behavior.	

5.1.3.74 nv set/unset interface acl outbound control-plane

	nv set interface <iface-id> acl <acl-id> onbound control-plane nv unset interface <iface-id> acl <acl-id> onbound control-plane Configure the binding of the given ACL on the specified interface. The unset form of the command removes the binding of the given ACL on the specified interface.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
	acl-id	ACL name
History	25.02.2002	
Example	`admin@nvos:~$ nv set interface eth0 acl acl-default-outbound outbound control-planeadmin@nvos:~$ nv unset interface eth0 acl acl-default-outbound outbound control-plane`	
REST API	PATCH https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/outbound/control-plane DELETE https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/outbound/control-plane	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	An ACL cannot be bound to inbound and inbound control-plane or cannot be bound to outbound and outbound control-plane on the same interface!	

5.1.3.75 nv set/unset interface acl outbound

	nv set interface <iface-id> acl <acl-id> outbound nv unset interface <iface-id> acl <acl-id> outbound Configure/remove the binding of the given ACL on the specified interface.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
	acl-id	ACL name
History	25.02.2002	
Example	`admin@nvos:~$ nv set interface eth0 acl acl-default-outbound outboundadmin@nvos:~$ nv unset interface eth0 acl acl-default-outbound outbound`	
REST API	PATCH/DELETE https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}/outbound	
Related Commands	nv set acl <acl-id> rule <rule-id>	
Notes	An ACL cannot be bound to inbound and inbound control-plane or cannot be bound to outbound and outbound control-plane on the same interface!	

5.1.3.76 nv set/unset interface acl

	nv set interface <iface-id> acl nv unset interface <iface-id> acl Set the ACL configurations for the particular interface. Unset the ACL configurations for the particular interface.

Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv set interface eth0 acl
	admin@nvos:~$ nv unset interface eth0 acl</pre>	

REST API	PATCH https://<ip>/nvue_v1/interface/{interface-id}/acl DELETE https://<ip>/nvue_v1/interface/{interface-id}/acl	
Related Commands		
Notes	Unsetting an interface ACL (at any level) should not restore the default binding; it should behave as a standard unset operation. Setting or unsetting rules on an interface works just like standard NVUE behavior.	

5.1.3.77 nv set/unset interface acl id

	nv set interface <id> acl <acl-id> nv unset interface <id> acl <acl-id> Set the ACL configurations for the particular interface acl-id. Unset the ACL configurations for the particular interface acl-id.	
Syntax Description	iface-id	Interface could be one of 'eth0' or 'loopback'
	acl-id	ACL Name
History	25.02.2002 25.02.70xx Updated ACL names	

Example	<pre>admin@nvos:~$ nv set interface eth0 acl acl-default-dos
	admin@nvos:~$ nv unset interface eth0 acl acl-default-whitelist</pre>	

REST API	PATCH https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id} DELETE https://<ip>/nvue_v1/interface/{interface-id}/acl/{acl-id}	
Related Commands		
Notes	Unsetting an interface ACL (at any level) should not restore the default binding; it should behave as a standard unset operation. Setting or unsetting rules on an interface works just like standard NVUE behavior.	

5.1.3.78 nv set/unset interface

	nv set interface nv unset interface Set the interface configurations. Unset and unset the interface configurations.
Syntax Description	N/A
History	25.02.2002

Example	admin@nvos:~$ nv set interface admin@nvos:~$ nv unset interface
REST API	PATCH https://<ip>/nvue_v1/interface DELETE https://<ip>/nvue_v1/interface
Related Commands	
Notes	Unsetting an interface ACL (at any level) should not restore the default binding; it should behave as a standard unset operation. Setting or unsetting rules on an interface works just like standard NVUE behavior.

5.1.3.79 nv action clear acl counters

	nv action clear acl counters Clear the ACL counters in the show command.
Syntax Description	N/A
History	25.02.2002
Example	admin@nvos:~$ nv action clear acl counters
REST API	POST https://<ip>/nvue_v1/acl
Related Commands	nv set acl <acl-id> rule <rule-id>
Notes	

5.1.3.80 nv set acl rule action set dscp

	nv set acl rule action set dscp Set DSCP value for packets.	
Syntax Description	acl-id	ACL ID to manipulate
	rule-id	Rule to configure dscp

	Dscp-value	It could be enum or an integer. Enums supported: • af11 • af12 • af13 • af21 • af22 • af23 • af31 • af32 • af33 • af41 • af42 • af43 • cs1 • cs2 • cs3 • cs4 • cs5 • cs6 • cs7 • be • ef Or an integer in the range [0,63]
History	25.02.4002	
Example	 admin@nvos:~$ nv set acl ACL1 rule 10 action set dscp ef	
REST API	PATCH https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/action/set	
Related Commands	nv show acl rule action	
Notes	Supported only for the management interface. Configurable only in inbound and outbound directions!	

5.1.3.81 nv set/unset system control-plane acl

	nv set system control-plane acl <acl-id><outbound | inbound> nv unset system control-plane acl <acl-id> Binds ACL to the system control-plane and sets its direction. The unset form of the command unbinds a specific system control-plane ACL. If no particular ACL ID is specified, the command will unbind all system control-plane ACLs.	
Syntax Description	acl-id	ACL ID
	inbound	Binds in inbound direction
	outbound	Binds in outbound direction
History	25.02.70xx	
Example	 admin@nvos:~$ nv set system control-plane acl acl-default-whitelist-ipv6 inbound admin@nvos:~$ nv unset system control-plane acl acl_1 admin@nvos:~$ nv unset system control-plane acl	

REST API	PATCH https://<ip>/nvue_v1/system/control-plane/acl/{acl-id}
Related Commands	nv show system control-plane acl
Notes	

5.1.3.82 nv show system control-plane acl

	nv show system control-plane acl Show system control-plane ACL configuration.
Syntax Description	N/A
History	25.02.70xx
Example	

```
admin@nvos:~$ nv show system control-plane acl
    ACL Name                    Rule ID  In Packets  In Bytes    Out Packets  Out Bytes
    --------------------------  -------  ----------  ----------  -----------  ---------
    acl-default-dos             10       3819        236.91 KB
                                20       1218        64.07 KB
                                30       0           0 Bytes
    acl-default-dos-ipv6        10       0           0 Bytes
                                20       0           0 Bytes
                                30       0           0 Bytes
                                40       0           0 Bytes
    acl-default-outbound        10                               0            0 Bytes
                                20                               295          32.96 KB
    acl-default-outbound-ipv6   10                               0            0 Bytes
                                20                               0            0 Bytes
    acl-default-whitelist       10       0           0 Bytes
                                20       0           0 Bytes
                                30       1083        82.80 KB
    acl-default-whitelist-ipv6  10       0           0 Bytes
                                20       0           0 Bytes
                                30       0           0 Bytes
```

REST API	GET https://<ip>/nvue_v1/system/control-plane/acl
Related Commands	nv set system control-plane acl nv show system control-plane
Notes	

5.1.3.83 nv show system control-plane

	nv show system control-plane Show system control-plane configuration.
Syntax Description	N/A
History	25.02.70xx

Example	admin@nvos:~$ nv show system control-plane acl ====== ACL Name Rule ID In Packets In Bytes Out Packets Out Bytes --- acl-default-dos 10 3819 236.91 KB 20 1218 64.07 KB 30 0 0 Bytes acl-default-dos-ipv6 10 0 0 Bytes 20 0 0 Bytes 30 0 0 Bytes 40 0 0 Bytes acl-default-outbound 10 0 0 Bytes 20 295 32.96 KB acl-default-outbound-ipv6 10 0 0 Bytes 20 0 0 Bytes acl-default-whitelist 10 0 0 Bytes 20 0 0 Bytes 30 1083 82.80 KB acl-default-whitelist-ipv6 10 0 0 Bytes 20 0 0 Bytes 30 0 0 Bytes
REST API	GET https://<ip>/nvue_v1/system/control-plane/
Related Commands	nv show system control-plane acl
Notes	

5.1.3.84 nv show system control-plane acl id

	nv show system control-plane acl <acl-id> Show system control-plane ACL identification.	
Syntax Description	acl-id	ACL ID
History	25.02.70xx	
Example	admin@nvos:~$ nv show system control-plane acl acl-default-whitelist-ipv6 Statistics ========== Rule In Packet In Byte Out Packet Out Byte Action Match --- 10 0 0 Bytes permit ip udp dest-port : 161 connection-state new : False established : False protocol : udp 20 0 0 Bytes permit ip	
REST API	GET https://<ip>/nvue_v1/system/control-plane/acl/{acl-id}	
Related Commands	nv set system control-plane acl nv show system control-plane acl	
Notes		

5.1.3.85 nv show system control-plane acl statistics

	nv show system control-plane acl <acl-id> statistics Show system control-plane ACL statistics configuration.	
Syntax Description	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show system control-plane acl acl-default-whitelist-ipv6 statistics Rule In Packet In Byte Out Packet Out Byte Action Match ---- --- --- --- --- --- 10 0 0 Bytes permit ip udp dest-port : 161 connection-state new : False established : False protocol : udp	
REST API	GET https://<ip>/nvue_v1/system/control-plane/acl/{acl-id}/statistics	
Related Commands		
Notes		

5.1.3.86 nv show system control-plane acl id statistics

	nv show system control-plane acl <acl-id> statistics <rule-id> Show system control-plane ACL ID statistics rule-id.	
Syntax Description	acl-id	ACL ID
	rule-id	Rule number (integer: 1–65535)
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface eth0 acl acl-default-whitelist statistics 10 operational applied --- match ip protocol udp udp [dest-port] 161 [connection-state] new [connection-state] established action permit inbound packet 0 byte 0 Bytes	
REST API	GET https://<ip>/nvue_v1/system/control-plane/acl/{acl-id}/statistics/{rule-id}	
Related Commands		
Notes		

5.1.3.87 nv show system control-plane acl inbound

	nv show system control-plane acl <acl-id> inbound Show system control-plane ACL IC inbound direction details.	
Syntax Description	acl-id	ACL ID
History	25.02.70xx	
Example	admin@nvos:~$ nv show system control-plane acl acl-default-whitelist-ipv6 inbound Statistics =============== Rule In Packet In Byte Action Match ---- ---------- ---------- ---------- ---------- 10 0 0 Bytes permit ip udp dest-port : 161 connection-state new : False established : False protocol : udp	
REST API	GET https://<ip>/nvue_v1/system/control-plane/acl/{acl-id}/inbound	
Related Commands		
Notes	If an ACL is configured in one direction but not the other, it will appear in the parent show command (nv show interface acl) and not in the show command for the direction where it is not configured.	

5.1.3.88 nv show system control-plane acl outbound

	nv show system control-plane acl <acl-id> outbound Show system control-plane ACL ID outbound direction details.
Syntax Description	N/A
History	25.02.70xx
Example	admin@nvos:~$ nv show system control-plane acl acl-default-outbound outbound Statistics =============== Rule Out Packet Out Byte Action Match ---- ---------- ---------- ---------- ---------- 10 33 1716 Bytes deny ip connection-state invalid : False 20 3876 237065 Bytes permit
REST API	GET https://<ip>/nvue_v1/system/control-plane/acl/{acl-id}/outbound
Related Commands	
Notes	If an ACL is configured in one direction but not the other, it will appear in the parent show command (nv show interface acl) and not in the show command for the direction where it is not configured.

5.1.3.89 nv set/unset acl rule action recent

	nv set acl rule action recent nv unset acl rule action Set ACL rule action recent. The unset form of the command removes ACL rule action recent.	
Syntax Description	acl-id	ACL ID
	rule-id	Rule number (integer: 1–65535)
History	25.02.70xx	
Example	`admin@nvos:~$ nv set acl acl_2 rule 1 action recent`	
REST API	GET https://<ip>/nvue_v1/acl/{acl-id}/rule/{rule-id}/action/recent	
Related Commands		
Notes	Only one action can be specified per rule. Any rule that matches the specified criteria will be accepted by the system. If a rule has no action defined, the default action will be "permit." If "match.ip.recent-list.action" is set, the default action is "recent." There is also the option to explicitly set the action to "recent" when "match.ip.recent-list" is configured.	

5.2 Attestation

For more information, see the following sections:

- TPM

5.2.1 TPM

TPM (Trusted Platform Module) is a hardware-based security technology that protects system integrity by securely storing cryptographic keys and measurements. It supports functionalities such as secure boot, attestation, and encryption.

NVOS measures the boot chain into TPM 2.0 PCR banks (SHA-384). Each PCR captures a specific stage of the boot process via extend operations.

- PCR 0 — Platform firmware (UEFI): measured by the UEFI firmware at power-on.
- PCR 4 — Boot loader chain: Includes Shim binary, GRUB binary, and kernel image. Measurements are performed sequentially: UEFI measures Shim, Shim measures GRUB, and GRUB measures the kernel..
- PCR 7 — Secure Boot policy: Includes the certificates used for verify the various secure boot components. Measured by the UEFI firmware and Shim.

5.2.1.1 TPM Commands

- TPM Commands

5.2.1.2 TPM Commands

5.2.1.2.1 nv action generate system security tpm

	nv action generate system security tpm <pcrs> <nonce> [algorithm] Generate quotes file.	
Syntax Description	pcrs	Platform Configuration Registers to be included in the quote <1-30>, divided by ",". Both quote and PCRs use the same hash algorithm.
	nonce	Hex string, up to 512 bits (128 hex letters)
	algorithm	Hashing algorithm to be used (e.g., sha384)
Default	Algorithm-sha384	
History	25.02.2002	
Example	`admin@nvos:~$ nv action generate system security tpm 1,2 12 algorithm sha384`	
REST API	POST https://<ip>/nvue_v1/system/security/tpm/quote	
Related Commands		
Notes		

5.2.1.2.2 nv action upload system security tpm

	nv action upload sys security tpm <file-name> <remote-url> Upload configuration file.	
Syntax Description	file-name	File to be uploaded (IAK.crt, quotes.json, or oIAK.crt). Note: quotes.json is a Base64-encoded JSON of quote.bin and signature, available after generating using nv action generate system security tpm.
	remote-url	Destination image file name Remote url path to upload a file to. Format: [protocol]://username[:password]@hostname/path/filename Supported protocols: SCP, FTP, SFTP, and HTTPS

Default	N/A
History	25.02.2002 25.02.4002 Added HTTPS support in remote-url
Example	admin@nvos:~$ nv action upload sys security tpm IAK.crt scp://user:pass@host/path/IAK.crt
REST API	POST https://<ip>/nvue_v1/system/security/tpm/upload
Related Commands	
Notes	

5.2.1.2.3 nv show system security tpm oiak

	nv show sys security tpm oiak Display owner IAK certificate.	
Syntax Description	N/A	
Default	N/A	
History	25.02.4282	
Example	admin@nvos:~$ nv show sys security tpm oiak operational ----- ----------- plain Certificate: Data: Version: 3 (0x2) … …	
REST API	GET https://<ip>/nvue_v1/system/security/tpm/oiak	
Related Commands	nv action import system security tpm oiak	
Notes		

5.2.1.2.4 nv action import system security tpm oiak

	nv action import system security tpm oiak Import system security TPM owner IAK.	
Syntax Description	remote-url	A local/remote URI from where the certificate file (containing the public-key) can be retrievied
	data	The raw data bytes (e.g., PEM string) of the certificate
Default	N/A	
History	25.02.4282	
Example	admin@nvos:~$ nv action import system security tpm oiak remote-url scp://u:p@s/oIAK.crt	
REST API	POST https://<ip>/nvue_v1/system/security/tpm/oiak	
Related Commands	nv action delete system security tpm oiak nv show sys security tpm oiak	

Notes	

5.2.1.2.5 nv action delete system security tpm oiak

	nv action delete system security tpm oiak Delete system security TPM owner IAK.	
Syntax Description	N/A	
Default	N/A	
History	25.02.4282	
Example	`admin@nvos:~$ nv action delete system security tpm oiak`	
REST API	DELETE https://<ip>/nvue_v1/system/security/tpm/oiak	
Related Commands	nv action import system security tpm oiak	
Notes		

5.3 Authentication Authorization and Accounting

AAA (authentication, authorization, and accounting) supports configuring local accounts and remote servers using protocols like RADIUS, TACACS+, and LDAP. The AAA object model includes user management, general configurations, and per-protocol settings.

AAA configuration can be viewed on NVOS.

```
admin@nvos:~$ nv show system aaa
```

AAA authentication consists of authentication order and authentication failthrough.

5.3.1 Authentication Order

Authentication order specifies the sequence of protocols (radius, tacacs, ldap, local) used for authentication, separated by spaces, for example:

```
admin@nvos:~$ nv set system aaa authentication order radius local
# or
admin@nvos:~$ nv set system aaa authentication order local ldap
```

Authentication order must include local and one of the following: radius, tacacs, or ldap.

5.3.2 Authentication Failthrough

Authentication failthrough defines the behavior of authentication when it is rejected locally or by an AAA server.
When authentication failthrough is disabled (default), the authentication process is blocked if the

user password is rejected either locally or by the AAA server.
When authentication failthrough is enabled, the authentication process continues to the next AAA server or method if it is rejected.

```
admin@nvos:~$ nv set system aaa authentication failthrough ?

    <arg>                  Configure failthrough.
                           "Enabled" login authentication continues to the next option on both server and
authentication errors.
                           "Disabled" login authentication continues only on server errors.
                            (enum:enabled, disabled | string | default:disabled)
```

Authentication failthrough does not impact behavior when the AAA server is unavailable. After a server timeout, the switch will try the next server or method in order.

For more information on Authentication Authorization and Accounting, see the following sections:

- User Accounts
- LDAP Authentication and Authorization
- TACACS
- RADIUS
- Role Based Access Control
- Authorization
- SSH for Remote Access

5.3.3 User Accounts

By default, NVOS has two user accounts: *admin* and *monitor*.

The *admin* account:

- The default password of the account is "admin".
- The system administrator account. It is part of *sudo* group and it has sudo privileges.
- The account has permissions to run any show, set or action commands.

The *monitor* account:

- The default password of the account is "monitor".
- The system monitor account. It has read-only privileges.
- The account has permissions to run show commands only.

5.3.3.1 First Login

User will be required to change the default passwords for admin and monitor accounts upon first login. The new password must comply with the default password hardening rules (see Password Hardening section).

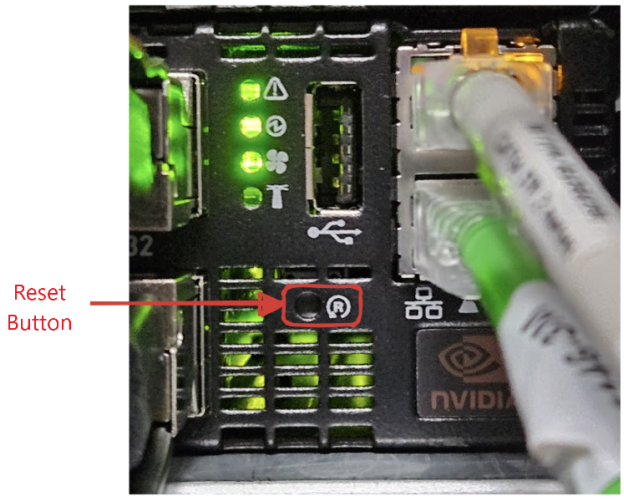
The new configured password is treated like any other configuration. If the configuration is not saved, the user will need to reconfigure it upon the system's next boot.

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




WARNING: Your password has expired.
You must change your password now!
New password:
Retype new password:
```

5.3.3.2 Reset Local Users' Passwords

If the passwords for the local user accounts on the switch are forgotten, a recovery method exists. Press and hold the Reset button for at least 15 seconds to reset the passwords for the 'admin' and 'monitor' accounts. This process also deletes any non-default users (i.e., users other than 'admin' and 'monitor'), resets the passwords for these default users, and expires them. The expiration forces a password change upon the first login.

Reset Button on the Physical Switch



The image is for illustration purposes only. The button's location may vary depending on the switch model.

5.3.3.3 User Account Commands

- User Account Commands

5.3.3.4 User Account Commands

5.3.3.4.1 nv show system aaa user

	nv show system aaa user Displays list of users, their role and status.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show system aaa user Username Full-name Role State -------- ------------------- ------- ------- admin System Administrator admin enabled monitor System Monitor monitor enabled
REST API	GET https://<ip>/nvue_v1/system/aaa/user
Related Commands	nv set system aaa user
Notes	

5.3.3.4.2 nv show system aaa user id

	nv show system aaa user <user-id> Displays configuration of a user.	
Syntax Description	user-id	The user (e.g., monitor, test)
Default	N/A	

History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system aaa user admin operational applied state enabled enabled role admin admin full-name System Administrator System Administrator password * * hashed-password * * ssh [authorized-key]
REST API	GET https://<ip>/nvue_v1/system/aaa/user/{user-id}
Related Commands	nv show system aaa user nv set system aaa user
Notes	

5.3.3.4.3 nv show system aaa user ssh authorized-key

	nv show system aaa user <user-id> ssh authorized-key <authorized-key-id> Lists all SSH keys associated with the specified user.	
Syntax Description	user-id	The user (e.g., monitor, test)
	authorized-key-id	Name of the SSH key (item-name) (enum: saved keys of user)
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output	
Example	admin@nvos:~$ nv show sys aaa user admin ssh authorized-key SSH Key Name Key string Key Type key1 * ecdsa-sha2-nistp256	
REST API	GET https://<ip>/nvue_v1/system/aaa/user/<user>/ssh/authorized-key	
Related ommands		
Notes	The key string is obfuscated yet not regarded as a secret within NVOS.	

5.3.3.4.4 nv show system aaa user ssh authorized-key id

	nv show system aaa user <user-id> ssh authorized-key <ssh-authorized-key-id> Displays details for a specific SSH key.	
Syntax Description	user-id	The user (e.g., monitor, test)
	authorized-key-id	Name of the SSH key (item-name) (enum: saved keys of user)
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output	

Example	admin@nvos:~$ nv show sys aaa user admin ssh authorized-key key1 operational applied ---- ------------------- ------------------- key * * type ecdsa-sha2-nistp256 ecdsa-sha2-nistp256
REST API	GET https://<ip>/nvue_v1/system/aaa/user/<user-id>/ssh/authorized-key/<ssh-authorized-key-id>
Related Commands	
Notes	The key string is obfuscated yet not regarded as a secret within NVOS.

5.3.3.4.5 nv show system aaa user ssh

	nv show system aaa user <user-id> ssh Display user SSH configuration.	
Syntax Description	user-id	The user (e.g., monitor, test)
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show sys aaa user admin ssh operational applied ---------------- ---------- ------- [authorized-key] key1 key1	
REST API	GET https://<ip>/nvue_v1/system/aaa/user/<user-id>/ssh	
Related Commands		
Notes	Key string in obfuscated but is not considered a secret in nvos.	

5.3.3.4.6 nv show system aaa allow-reset-local-passwords

	nv show system aaa allow-reset-local-passwords Displays state of resetting the local users passwords upon long reboot press
Syntax	N/A
Default	Enabled
History	25.02.2002
Example	admin@nvos:~$ nv show system aaa allow-reset-local-passwords operational applied ----- ---------- ------- state enabled enabled
REST API	GET https://<ip>/nvue_v1/system/aaa/allow-reset-local-passwords
Related Commands	nv set system allow-reset-local-passwords state <enabled|disabled>
Notes	

5.3.3.4.7 nv show system aaa user spiffe-id

	nv show system aaa user <user-id> spiffe-id Display all SPIFFE IDs of a given user.
Syntax	N/A
Default	Enabled
History	25.02.4002
Example	admin@nvos:~$ nv show sys aaa user admin ssh spiffe-id Spiffe Id ---------- spiffe://example.com/prod
REST API	GET https://<ip>/nvue_v1/system/aaa/user/<user>/spiffe-id
Related Commands	nv set system aaa user <user-id> spiffe-id <spiffe-id>
Notes	Same as with certificate, the value for each spiffe-id is empty.

5.3.3.4.8 nv set/unset system aaa user ssh authorized-key

	nv set system aaa user <user-id> ssh authorized-key <ssh-authorized-key-id> {key | type} nv unset system aaa user <user> ssh authorized-key <ssh-authorized-key-id> {key | type} Authorized SSH key configuration. The unset form of the command clears configuration of SSH parameters for a user.	
Syntax Description	user-id	Name of the user (user-name) (enum: local users)
	authorized-key-id	Name of the SSH key (item-name) (enum: saved keys of user)
	key	The base64 contents of the key (key-string)
	type	The type of encoded key (string | enum:ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, ssh-ed25519, ssh-rsa | default:ssh-rsa)
Default	key=N/A type=ssh-rsa	
History	25.02.2002	
Example	admin@nvos:~$ nv set sys aaa user admin ssh authorized-key key1 key AAAdnfs…== admin@nvos:~$ nv set sys aaa user admin ssh authorized-key key1 type nistp384 admin@nvos:~$ nv unset sys aaa user admin ssh authorized-key admin@nvos:~$ nv unset sys aaa user admin ssh authorized-key key1 admin@nvos:~$ nv unset sys aaa user admin ssh authorized-key key1 type admin@nvos:~$ nv unset sys aaa user admin ssh authorized-key key1 key	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/user/<user-id>/ssh/authorized-key/<ssh-authorized-key-id>	
Related Commands		
Notes		

5.3.3.4.9 nv set/unset system aaa user

	nv set system aaa user <user-id> nv unset system aaa user <user-id> Specifies a username and creates a user account. New users are created initially with admin privileges. The unset form of the command deletes the user account.	
Syntax Description	user-id	The user. Username max length is 32 and it begins with a letter or an underscore, followed by letters, digits, underscores, or dashes. They can end with a dollar sign.
Default	The following usernames are available by default: * admin * monitor	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa user test`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/user/{user-id}	
Related Commands	nv show system aaa user nv set system aaa user password	
Notes	• New users must have a password. • Default users cannot be deleted.	

5.3.3.4.10 nv set/unset system aaa user full-name

	nv set system aaa user <user-id> full-name <full-name> nv unset system aaa user <user-id> full-name <full-name> Configures user's full-name (Gecos Field). The unset form of the command sets user full-name (Gecos Field) to empty.	
Syntax Description	user-id	The user
	full-name	The full name of the user
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa user test full-name "Test User"`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/user/{user-id}	
Related Commands	nv show system aaa user nv set system aaa user	
Notes		

5.3.3.4.11 nv set/unset system aaa user state

	nv set system aaa user <user-id> state <enable | disable> nv unset system aaa user <user-id> state Enables/disables the user account. The unset form of the command returns the user account state to its default state (enabled).	
Syntax Description	user-id	The user
Default	Enabled	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa user test state disabled`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/user/{user-id}	
Related Commands	nv show system aaa user nv set system aaa user	
Notes	Disabling a user account will terminate all user bash terminals.	

5.3.3.4.12 nv set/unset system aaa user role

	nv set system aaa user <user-id> role <role-id> nv unset system aaa user <user-id> role Configures user role (capabilities). The unset form of the command return the user account role to its default (admin).	
Syntax Description	user-id	The user
	role-id	The name of the role
Default	admin	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa user test role monitor`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/user/{user-id}	
Related Commands	nv show system aaa user nv show system aaa role nv set system aaa user	
Notes		

5.3.3.4.13 nv set/unset system aaa user password

	nv set system aaa user <user-id> password <password> nv unset system aaa user <user-id> password Configures a login password in cleartext. The unset form of the command clears the user password for non-default users. For default users, the default password will be expired and must be reconfigured in the next login.

Syntax Description	user-id	The user

	password	A password for the user in string form.
		A string containing special Linux characters must be quoted or have the special characters escaped (i.e., add "\" before each special character).
		Examples:
		`pass\!word`
		`"pass!word"`
		
		A leading dot is a special case and it must be escaped even if it is quoted:
		Examples:
		`"\.password"`
		`\\.password`

Default	N/A	
History	25.02.2002	
Example	 ``` admin@nvos:~$ nv set system aaa user test password "pa\!sswOrd" admin@nvos:~$ nv set system aaa user test password Enter new password: Confirm password: ```	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/user/{user-id}	
Related Commands	nv show system aaa user nv set system aaa user	

Notes	• If no password was specified, user will be prompted to configure the password.
	• If password hardening is enabled, the new password must match all configured policies.
	• A string containing special Liznux characters must be quoted or have the special characters escaped (i.e., add "\" before each special character).
	Examples:
	`pass\!word`
	`"pass!word"`
	
	A leading dot is a special case and it must be escaped even if it is quoted:
	Examples:
	`"\.password"`
	`\\.password`
	• A password is required, therefore a password must be configured before applying new configurations.	

5.3.3.4.14 nv set/unset system aaa user hashed-password

	nv set system aaa user <user-id> hashed-password <hashed-password> nv unset system aaa user <user-id> hashed-password Configures a login password in encrypted format. The unset form of the command clears the user hashed-password.	
Syntax Description	user-id	The user

	hashed-password	A password for the user in encrypted text. Special Linux characters must be escaped (add "\" before each special character).
Default	N/A	
History	25.02.2002	
Example	 admin@nvos:~$ nv set system aaa user test password "\$y\$j9T\$YwHwJEhi5c2oCgNNVJZgR0\$TboB1DcoS2iGmneLa/9Y54hsAgq9milQKGYmmkRffJC"	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/user/{user-id}	
Related Commands	nv show system aaa user nv set system aaa user nv set system aaa user password	
Notes	• If password hardening is enabled, hashed-password configuration will be blocked. • Special Linux characters must be escaped (i.e., add "\" before each special character). • A password is required, therefore a password must be configured before applying new configurations.	

5.3.3.4.15 nv set/unset system aaa allow-reset-local-passwords state

	nv set system aaa allow-reset-local-passwords state <enabled | disabled> nv unset system aaa allow-reset-local-passwords state Enables/disables the ability to reset local users' passwords upon long reboot press. The unset form of the command returns the state of feature to its default state (enabled).	
Syntax	state	enabled, disabled
Default	Enabled	
History	25.02.2002	
Example	 admin@nvos:~$ nv set system aaa allow-reset-local-passwords state disabled	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/allow-reset-local-passwords	
Related Commands	nv show system aaa allow-reset-local-passwords	
Notes		

5.3.3.4.16 nv set/unset system aaa user spiffe-id

	nv set system aaa user <user-id> spiffe-id <spiffe-id> nv unset system aaa user <user-id> spiffe-id Configure all SPIFFE IDs of a given user. The unset form of the command clears all SPIFFE IDs for a user.	
Syntax Description	spiffe-id	SPIFFE ID mapped to the user (string) SPIFFE ID must be in format `spiffe://trust-domain/path`, where the trust-domain and path has only letters, numbers, dots, dashes, and underscores

Default	N/A
History	25.02.4002
Example	`admin@nvos:~$ nv set system aaa user admin spiffe-id spiffe://trust/domain`
REST API	PATCH https://<ip>/nvue_v1/system/aaa/user/<user>/ spiffe-id /<ssh- spiffe-id>
Related Commands	nv show system aaa user spiffe-id
Notes	

5.3.4 LDAP Authentication and Authorization

NVOS implements LDAP client AAA (Accounting, Authentication, and Authorization) in a transparent way with minimal configuration. There is no need to create accounts or directories on the switch.

NVOS uses Pluggable Authentication Modules (PAM) and Name Service Switch (NSS) for user authentication. NSS enables PAM to use LDAP to provide user authentication, group mapping, and information for other services on the system.

5.3.4.1 Supported Features

- Authentication using PAM: Supports login, SSH, **sudo**, and **su**.
- Runs over the eth0 management interface.
- Supports up to eight LDAP servers.

5.3.4.2 LDAP Configuration

LDAP configuration consists of two levels:

1. Global Configuration: Settings that apply to all LDAP servers unless overridden.
2. Per-Server Configuration: Specific settings for individual LDAP servers.

If a per-server configuration is not defined, the system will automatically use the settings from the global configuration.

All nv ldap commands are in LDAP Commands section. Global commands are under `/system/aaa/ldap`, and per-server commands are under `/system/aaa/ldap/server/<hostname-id>`.

5.3.4.3 LDAP Groups and User Privileges

The LDAP server configuration assigns them to specific groups. The GID is now unique by default and can be assigned to any GID by design.

1. Admin privileged users: 4(adm), 27(sudo), 999(docker), 1001(redis), 995(nvaction), 997(nvapply)
2. Monitor privileged users: 4(adm), 998(nvshow)
3. Non-privileged users: No NV commands access

5.3.4.3.1 LDAP Server Group Configuration Example

Below is an example of configuring LDAP server groups. This configuration allows you to define a group of LDAP servers with common settings while enabling server-specific overrides when necessary.

```
dn: cn=nvset,ou=People,dc=itzgeek,dc=local
objectClass: posixGroup
cn: nvaction
gidNumber: 995
memberUid: adminuser
```

5.3.4.4 LDAP Secure Connection

The SSL section enables configuring the encryption mode for the LDAP client to ensure secure communication.

- Supported Encryption Modes: `start-tls`, `ssl`.
- Default CA Certificate Bundle: The LDAP client uses the default CA certificate bundle located at `/etc/ssl/certs/ca-certificates.crt`.
- Certificate Validation: Certificate validation may be skipped using SSL settings `cert-verify`. When certificate validation is skipped, the certificate is used only to establish a secure connection, without verifying its authenticity.

Ensure proper configuration to maintain secure and reliable LDAP connections.

5.3.4.5 LDAP Client Simple Configuration Example

Below is a simple example of configuring an LDAP client. This setup includes basic global settings and per-server configuration.

```
admin@nvos:~$ nv set system aaa ldap bind-dn <ldap-server-bind-dn>
admin@nvos:~$ nv set system aaa ldap secret "ldap-secret"
admin@nvos:~$ nv set system aaa ldap server <ldap-server-ip>
admin@nvos:~$ nv set system aaa ldap base-dn <ldap-server-base-dn>
# set global aaa configs
admin@nvos:~$ nv set system aaa authentication order ldap local
admin@nvos:~$ nv config apply -y
```

NVOS uses Pluggable Authentication Modules (PAM) and Name Service Switch (NSS) for user authentication. NSS enables PAM to use LDAP to provide user authentication, group mapping, and information for other services on the system.

- NSS specifies the order of the information sources that resolve names for each service. Using NSS with authentication and authorization provides the order and location for user lookup and group mapping on the system.
- PAM handles the interaction between the user and the system, providing login handling, session setup, authentication of users, and authorization of user actions.

LDAP authentication is sensitive to network delay. For optimal performance, NVIDIA recommends a round trip time of 10ms or less between LDAP clients and the LDAP server. If latency is between 10-50ms, NVIDIA recommends changing the authentication order to prioritize local authentication before LDAP. For connections exceeding 50ms of latency, authentication might experience unacceptable delays; consider alternative authentication methods.

5.3.4.6 Configure LDAP Server Settings

LDAP server settings can be configured with NVUE.

5.3.4.6.1 Connection

Configure the following connection settings:

- The hostname or IP address of the LDAP server from which you want to import users. If you use multiple LDAP servers, you can also set a priority for each server.
- The port number of the LDAP server if you are using a non-default port. The default port number for LDAP is TCP and UDP port 389.
- Authenticated (Simple) BIND credentials. The BIND credentials are optional; if you do not specify the credentials, the switch assumes an anonymous bind. To use SASL (Simple Authentication and Security Layer) BIND, which provides authentication services using other mechanisms such as Kerberos, contact your LDAP server administrator for authentication information.

The following example configures the LDAP server and port, and the BIND credentials.

```
admin@nvos:~$ nv set system aaa ldap server ldapserver1
admin@nvos:~$ nv set system aaa ldap port 388
```

```
admin@nvos:~$ nv set system aaa ldap bind-dn CN=nvset-admin,CN=Users,DC=rtp,DC=example,DC=test
admin@nvos:~$ nv set system aaa ldap secret 1Q2w3e4r!
admin@nvos:~$ nv config apply
```

The following example sets the priority to 2 for ldapserver2 when using multiple LDAP servers:

```
admin@nvos:~$ nv set system aaa ldap server ldapserver2 priority 2
```

5.3.4.6.2 Set the Authentication Order

To prioritize the order in which NVOS attempts different authentication methods to verify user access to the switch, you set the authentication order. By default, NVOS verifies users according to their local passwords.

When AAA authentication failthrough is enabled, if you set the authentication order to start with LDAP, but the LDAP servers do not have the user in the directory or does not respond, NVOS tries local password authentication.

If no LDAP server is available, local authentication is applied.

To set the authentication order to start with LDAP before local authentication:

```
admin@nvos:~$ nv set system aaa authentication order ldap local
admin@nvos:~$ nv config apply
```

5.3.4.6.3 Search Function

When an LDAP client requests information about a resource, the client must connect and bind to the server, then perform one or more resource queries depending on the lookup. All search queries to the LDAP server use the configured search *base*, *filter*, and the desired entry (*uid=myuser*). If the LDAP directory is large, this search takes a long time. Define a more specific search base for the common *maps* (*passwd* and *group*).

```
admin@nvos:~$ nv set system aaa ldap base-dn ou=support,dc=rtp,dc=example,dc=test
admin@nvos:~$ nv config apply
```

5.3.4.6.4 LDAP Version

NVOS uses LDAP version 3 by default. If you need to change the LDAP version to 2:

```
admin@nvos:~$ nv set system aaa ldap version 2
admin@nvos:~$ nv config apply
```

5.3.4.6.5 LDAP Timeouts

NVOS provides two timeout settings:

- The bind timeout sets the number of seconds before the BIND operation times out. The default setting is 5 seconds.
- The search timeout sets the number of seconds before the search times out. The default setting is 5 seconds.

The following example sets both the BIND session timeout and the search timeout to 60 seconds.

```
admin@nvos:~$ nv set system aaa ldap timeout-bind 60
admin@nvos:~$ nv set system aaa ldap timeout-search 60
admin@nvos:~$ nv config apply
```

5.3.4.6.6 SSL Options

You can configure the following SSL options:

- SSL mode. You can specify, `none`, `ssl`, or `start-tls`.
- SL port.
- SSL certificate validation.
- SSL cipher suites. You can specify TLS1.2, TLS1.3, TLS-CIPHERS, or all.
- SSL CRL check.

To use IPv6 with SSL or Start TLS, you must set the hostname of the LDAP server instead of the IPv6 address with the nv set system aaa ldap server <hostname> command. See Configure LDAP Server Settings.

The following example sets the SSL mode to SSL, the port to 8443, enables the SSL certificate checker, sets the SSL cipher suites to TLS1.3 and the Certificate Authorities List to /etc/ssl/certs/ca-certificates.crt

```
admin@nvos:~$ nv set system aaa ldap ssl mode ssl
admin@nvos:~$ nv set system aaa ldap ssl port 8443
admin@nvos:~$ nv set system aaa ldap ssl cert-verify enabled
admin@nvos:~$ nv set system aaa ldap ssl tls-ciphers TLS1.3
admin@nvos:~$ nv config apply
```

5.3.4.6.7 Show LDAP Settings

To show the LDAP configuration settings on the switch, run the following commands:

- nv show system aaa ldap shows all the LDAP configuration settings.
- nv show system aaa ldap server shows the configured LDAP servers and their priorities.
- nv show system aaa ldap server <server-id> shows the priority for the specified LDAP server.
- nv show system aaa ldap ssl shows the LDAP SSL configuration settings.

The following example shows all the LDAP configuration settings:

```
admin@nvos:~$ nv show system aaa ldap
                operational                                          applied
--------------  ---------------------------------------------------  ----------
bind-dn         CN=nvset-admin,CN=Users,DC=rtp,DC=example,DC=test  CN=nvset-
admin,CN=Users,DC=rtp,DC=example,DC=test
base-dn         ou=users,dc=example,dc=com                           ou=users,dc=example,dc=com
port            389                                                  388
timeout-bind    5                                                    5
timeout-search  5                                                    5
secret          $nvsec$4fb719e24167246947d4f746f58696fc              $nvsec$4fb719e24167246947d4f746f58696fc
version         3                                                    3
[server]        ldapserver1                                          ldapserver1
ssl
  mode          ssl                                                  ssl
  port          8443                                                 8443
  cert-verify   enabled                                              enabled
  tls-ciphers   TLS1.3                                               TLS1.3
...
```

The following example shows the configured LDAP servers and their priorities:

```
admin@nvos:~$ nv show system aaa ldap server
```

```
Hostname     Priority
-----------  --------
ldapserver1  1
ldapserver2  2
```

The following example shows the SSL configuration settings:

```
admin@nvos:~$ nv show system aaa ldap ssl
            operational                      applied
----------  -------------------------------  ---------------------------------
mode        ssl                              ssl
port        8443                             8443
cert-verify  enabled                         enabled
tls-ciphers  TLS1.3                          TLS1.3
```

5.3.4.7 LDAP Verification Tools

The LDAP client daemon retrieves and caches password and group information from LDAP. To verify the LDAP interaction, use these command-line tools to trigger an LDAP query from the device.

5.3.4.7.1 Identify a User with the id Command

The `id` command performs a username lookup by following the lookup information sources in NSS for the *passwd* service. This returns the user ID, group ID and the group list retrieved from the information source. In the following example, the user *admin* is locally defined in `/etc/passwd`, and *myuser* is on LDAP. The NSS configuration has the `passwd` map configured with the sources `compat ldap`:

```
admin@radmin:~$ id admin
uid=1000(admin) gid=1000(admin)
groups=1000(admin),4(adm),27(sudo),999(docker),1001(redis),996(nvaction),993(nvapply)
admin@nvos:~$ id myuser
uid=1230(myuser) gid=3000(Development) groups=3000(Development),500(Employees),27(sudo)
```

5.3.4.7.2 getent

The `getent` command retrieves all records found with NSS for a given map. It can also retrieve a specific entry under that map. You can perform tests with the `passwd`, `group`, `shadow`, or any other map in the `/etc/nslcd.conf` file. The output from this command formats according to the map requested. For the `passwd` service, the structure of the output is the same as the entries in `/etc/passwd`. The group map outputs the same structure as `/etc/group`.

In this example, looking up a specific user in the passwd map, the user *admin* is locally defined in `/etc/passwd`, and *myuser* is only in LDAP.

```
admin@radmin:~$ getent passwd admin
admin:x:1000:1000:System Administrator:/home/admin:/bin/bash
admin@nvos:~$ getent passwd myuser
myuser:x:1230:3000:My Test User:/home/myuser:/bin/bash
```

In the next example, looking up a specific group in the group service, the group *admin* is locally defined in `/etc/groups`, and *netadmin* is on LDAP.

```
admin@nvos:~$ getent group admin
admin:x:1000:
admin@nvos:~$ getent group netadmin
```

```
netadmin:*:502:larry,moe,curly,shemp
```

Running the command getent passwd or getent group without a specific request returns all local and LDAP entries for the *passwd* and *group* maps.

5.3.4.8 LDAP Commands

- LDAP Commands

5.3.4.9 LDAP Commands

5.3.4.9.1 nv show system aaa ldap

	nv show system aaa ldap Show LDAP configurations.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output

Example	admin@nvos:~$ nv show system aaa ldap operational applied ---------- ---------- ---------- bind-dn cn=ldapadm,dc=itzgeek,dc=local cn=ldapadm,dc=itzgeek,dc=local base-dn dc=itzgeek,dc=local dc=itzgeek,dc=local port 389 389 timeout-bind 5 5 timeout-search 5 5 secret * * version 3 3 [server] 10.237.0.86 10.237.0.86 ssl mode none none port 636 636 cert-verify disabled disabled tls-ciphers all all filter passwd (objectClass=posixAccount) (objectClass=posixAccount) group (objectClass=posixGroup) (objectClass=posixGroup) shadow (objectClass=shadowAccount) (objectClass=shadowAccount) map passwd uid cn cn uidnumber gidnumber userpassword group cn memberuid member member gidnumber
REST API	GET https://<ip>/nvue_v1/system/aaa/ldap
Related Commands	nv set system aaa ldap
Notes	LDAP feature in NVOS, the switch is basicly an LDAP client that can be bind to an LDAP server, to support authentication to the switch via LDAP server instead local.

5.3.4.9.2 nv show system aaa ldap server

	nv show system aaa ldap server Show remote LDAP servers.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	admin@nvos:~$ nv show system aaa ldap server Hostname Priority ---------- -------- 10.237.0.86 1	
REST API	GET https://<ip>/nvue_v1/system/aaa/ldap/server	
Related Commands	nv set system aaa ldap server	
Notes	Show LDAP configured servers.	

5.3.4.9.3 nv set system aaa ldap server id

	nv set system aaa ldap server<server-id> Configure remote LDAP servers.	
Syntax Description	server-id	LDAP server ID: ipv4, ipv4-unicas, idn-hostname, ipv6
Default	N/A	

History	25.02.2002 25.02.3000 Updated command syntax
Example	`admin@nvos:~$ nv set system aaa ldap server 1.2.3.4`
REST API	SET https://<ip>/nvue_v1/system/aaa/ldap/server/<server-id>
Related Commands	nv show system aaa ldap server <server-id> nv show system aaa ldap server nv show system aaa ldap
Notes	

5.3.4.9.4 nv set system aaa ldap base-dn

	nv set system aaa ldap base-dn <base-dn> This command set the base-dn of the LDAP server.	
Syntax Description	base-dn	Configure base DN (Distinguished Name)
Default	ou=users dc=example dc=com	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa ldap base-dn "dc=itzgeek,dc=local"`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/base-dn	
Related Commands	nv show system aaa ldap	
Notes	A base dn is the point from where a server will search for users.	

5.3.4.9.5 nv set system aaa ldap bind-dn

	nv set system aaa ldap bind-dn <bind dn> This command sets the bind-dn of the ldap server.	
Syntax Description	bind dn	Configure bind DN (Distinguished Name)
Default	None	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa ldap bind-dn "cn=ldapadm,dc=itzgeek,dc=local"`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/bind-dn	
Related Commands	nv show system aaa ldap	
Notes	The Bind DN is the username that will be used to do the searching and request the authentication.	

5.3.4.9.6 nv set system aaa ldap port

	nv set system aaa ldap port <1–65535> Set LDAP authentication port.	
Syntax Description	port	Integer: 1–65535
Default	389	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa ldap port 389	
REST API	SET https://<ip>/nvue_v1/system/aaa/ldap/port	
Related Commands	nv show system aaa ldap	
Notes		

5.3.4.9.7 nv set system aaa ldap timeout-bind

	nv set system aaa ldap timeout-bind <seconds> Set global LDAP max wait until bind timeout (seconds).	
Syntax Description	Seconds	Number of seconds
Default	5	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa ldap timeout-bind 5	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/timeout-bind	
Related Commands	nv show system aaa ldap	
Notes		

5.3.4.9.8 nv set system aaa ldap timeout-search

	nv set system aaa ldap timeout-search <seconds> Set global LDAP max wait until search timeout (seconds).	
Syntax Description	Seconds	Number of seconds
Default	cn	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa ldap timeout-search 5	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/timeout-search	
Related Commands	nv show system aaa ldap	
Notes		

5.3.4.9.9 nv set system aaa ldap secret

	nv set system aaa ldap secret <secret-value> Set global LDAP server secret in cleartext.	
Syntax Description	secret value	Secret string
Default	3	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa ldap password 123asd	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/secret	
Related Commands	nv show system aaa ldap	
Notes		

5.3.4.9.10 nv set system aaa ldap map group cn

	nv set system aaa ldap map group cn <cn-str> Set LDAP search map for cn attribute for group database.	
Syntax Description	cn-str	Common name (string)
Default	None	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa ldap map group cn itzgeek	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/map/group/cn	
Related Commands	nv show system aaa ldap group	
Notes		

5.3.4.9.11 nv set system aaa ldap map group gidnumber

	nv set system aaa ldap map group gidnumber <gidnumber> Set LDAP search map for gidNumber attribute for group database.	
Syntax Description	gidnumber	gidNumber string
Default	None	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa ldap map group gidNumber 1000	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/map/group/gidnumber	
Related Commands	nv show system aaa ldap group	
Notes		

5.3.4.9.12 nv set system aaa ldap map group memberuid

	nv set system aaa ldap map group memberuid <memberuid> Set LDAP search map for memberUid attribute for group database.	
Syntax Description	memberuid	membeUid string
Default	None	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa ldap map group memberuid admingroup`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/map/group/memberuid	
Related Commands	nv show system aaa ldap group	
Notes		

5.3.4.9.13 nv set/unset system aaa ldap map passwd gidnumber

	nv set system aaa ldap map passwd gidnumber <gidnumber> Set LDAP map for gidNumber attribute for passwd database.	
Syntax Description	gidnumber	gidNumber string
Default	None	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa ldap map group gidnumber 1000`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/map/group/gidNumber	
Related Commands	nv show system aaa ldap passwd	
Notes		

5.3.4.9.14 nv set system aaa ldap map passwd uid

	nv set system aaa ldap map group uid <uid> Set LDAP map for UID attribute for passwd database.	
Syntax Description	uid	uid string
Default	None	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa ldap map group uid 1000`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/map/group/uid	
Related Commands	nv show system aaa ldap passwd	
Notes		

5.3.4.9.15 nv set system aaa ldap map passwd uidnumber

	nv set system aaa ldap map group uidnumber <uidnumber> Set LDAP map for uidNumber attribute for passwd database.	
Syntax Description	uidnumber	uidNumber string
Default	None	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa ldap map group uidnumber 1000`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/map/group/uidNumber	
Related Commands	nv show system aaa ldap passwd	
Notes		

5.3.4.9.16 nv set system aaa ldap map passwd userpassword

	nv set system aaa ldap map passwd userpassword <userpassword> Set LDAP map for userPassword attribute for passwd database.	
Syntax Description	userpassword	userpassword string
Default	None	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa ldap map group userpassword password`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/map/group/userpassword	
Related Commands	nv show system aaa ldap passwd	
Notes		

5.3.4.9.17 nv set system aaa ldap version

	nv set system aaa ldap version <ldap-version> Set LDAP protocol version to be used.	
Syntax Description	ldap-version	2 or 3
Default	3	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa ldap version 2`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap	
Related Commands	nv show system aaa ldap	
Notes		

5.3.4.9.18 nv set system aaa ldap ssl mode

	nv set system aaa ldap ssl mode <ssl-mode> Set the security mode used for LDAP server communication.	
Syntax Description	ssl-mode	none, ssl, start-tls
Default	None	
History	25.02.2002	
Example	`admin@nvos:~$ nv set sys aaa ldap ssl mode start-tls`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/ssl/mode	
Related Commands	nv show system aaa ldap	
Notes		

5.3.4.9.19 nv set system aaa ldap ssl cert-verify

	nv set system aaa ldap ssl cert-verify <enable | disable> Set CA certificate validation state.	
Syntax Description	enable	Validates certificate
	disable	Skips certificate validation
Default	Enabled	
History	25.02.2002	
Example	`admin@nvos:~$ nv set sys aaa ldap ssl mode cert-verify`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/ssl/cert-verify	
Related Commands	nv show system aaa ldap	
Notes		

5.3.4.9.20 nv set system aaa ldap ssl port

	nv set system aaa ldap ssl port <1–65535> Set LDAP's authentication port.	
Syntax Description	port	Integer: 1–65535
Default	636	
History	25.02.2002	
Example	`admin@nvos:~$ nv set sys aaa ldap ssl port 636`	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/ldap/ssl/port	
Related Commands	nv show system aaa ldap	

Notes	

5.3.5 TACACS

NVOS implements TACACS+ client AAA in a transparent way with minimal configuration. The client implements the TACACS+ protocol as described in this IETF document. There is no need to create accounts or directories on the switch. Accounting records go to all configured TACACS+ servers by default.

TACACS+ in NVOS:

- Uses PAM authentication and includes **login**, **ssh**, **sudo** and **su**.
- Allows users with privilege level 15 to run any command with sudo.
- Allows users with privilege level 15 to run NVUE **nv set**, **nv unset**, and **nv apply** commands in addition to **nv show** commands. TACACS+ users with a privilege level 7 can only execute **nv show** commands. Other privilege users have no permissions.
- Supports up to eight TACACS+ servers. Be sure to configure your TACACS+ servers in addition to the TACACS+ client. Refer to your TACACS+ server documentation.

5.3.5.1 Supported Features

- Authentication using PAM: includes **login**, **ssh**, **sudo** and **su**
- Runs over the eth0 management interface
- Up to eight TACACS+ servers

TACACS configuration is made of global configurations and per-server configurations. In general, if per-server configuration is not defined, the configuration will be taken from the global configuration.

All nv tacacs commands can be found in TACACS Commands, where global ones are direct under **/system/aaa/tacacs** and per-server ones or under **/system/aaa/tacacs/hostname/<hostname-id>**.

5.3.5.2 TACACS Users

NVOS supports three types of RADIUS users defined by priv-lvl configured in TACACS server.

- priv-lvl=15 # admin privileged users (nv set, nv config apply)
- priv-lvl=7 # monitor privileged users (nv show)
- priv-lvl=1 # non-privileged users (no nv commands access

5.3.5.2.1 TACACS Server Setup and Usage Example

TACACS server can be configured either on a remote host or on the switch itself (for testing or sanity-check).

Basic configuration for users and clients can be done in `/etc/tacplus_nss.conf` file.

Users Configuration

```
user = username {
        login = cleartext "login_password"
        pap = cleartext "pap_password"
        service = exec {
        priv-lvl=<15,7,1>
        }
}
```

Client Configuration

Client configuration allows specific client IPs and CIDR blocks.

```
key = "client-secret"
and:
acl = default   {
                #permit = 192\.168\.0\.
                permit = 10\.7\.140\.30
                permit = .*
}
```

After configuring a tacacs server, configure the client:

```
admin@nvos:~$ nv set system aaa tacacs hostname <tacacs-server-ip> secret tacacs-secret
admin@nvos:~$ nv set system aaa authentication order tacacs local
admin@nvos:~$ nv config apply -y
```

5.3.5.3 TACACS+ Accounting

5.3.5.3.1 TACACS Accounting Configuration

TACACS accounting logs user activity and commands executed on the system, providing an audit trail for security and compliance. It ensures accountability by sending these logs to configured TACACS+ servers. The logs will be sent to the first server to respond.

TACACS accounting is managed under the `/etc/tacplus_nss.conf` file.

After configuring a TACACS server and client, enable accounting with the command `nv set system aaa tacacs accounting state enabled`.

5.3.5.4 Required TACACS+ Client Configuration

Configure the following required settings on the switch (the TACACS+ client).

- Set the IP address or hostname of at least one TACACS+ server.
- Set the secret (key) shared between the TACACS+ server and client.

If you use NVUE commands to configure TACACS+, you must also set the priority for the authentication order for local and TACACS+ users.

NVUE commands require you to specify the priority for each TACACS+ server. You must set a priority even if you only specify one server.

The following example commands set:

- The TACACS+ server priority to 5.
- The IP address of the server to 192.168.0.30.
- The secret to abcdefghijklmnopqrstuvwxyz.

If you include special characters in the password (such as $), you must enclose the password in single quotes (').

- The authentication order so that TACACS+ authentication has priority over local (the lower number has priority).

```
admin@nvos:~$ nv set system aaa tacacs server 192.168.0.30 priority 5
admin@nvos:~$ nv set system aaa tacacs server 192.168.0.30 secret abcdefghijklmnopqrstuvwxyz
admin@nvos:~$ nv set system aaa authentication order tacacs local
admin@nvos:~$ nv config apply
```

If you want the server to use IPv6, you must add the `nv set system aaa tacacs server <server-id> prefer-ip-version 6` command:

```
admin@nvos:~$ nv set system aaa tacacs server SERVER1 priority 5
admin@nvos:~$ nv set system aaa tacacs server SERVER1 prefer-ip-version 6
...
```

If you configure more than one TACACS+ server, you need to set the priority for each server. If the switch cannot establish a connection with the server that has the highest priority, it tries to establish a connection with the next highest priority server. The server with the lower number has the higher priority. In the example below, server 192.168.0.30 with a priority value of 5 has a higher priority than server 192.168.1.30, which has a priority value of 8. NVOS allows to configure up to 8 servers with unique priority 1 to 8.

```
admin@nvos:~$ nv set system aaa tacacs server 192.168.0.30 priority 5
admin@nvos:~$ nv set system aaa tacacs server 192.168.0.30 secret abcdefghijklmnopqrstuvwxyz
admin@nvos:~$ nv set system aaa tacacs server 192.168.1.30 priority 8
admin@nvos:~$ nv set system aaa tacacs server 192.168.0.30 secret zyxwvutsrqponmlkjihgfedcba
admin@nvos:~$ nv config apply
```

5.3.5.5 Optional TACACS+ Configuration

You can configure the following optional TACACS+ settings:

- The port to use for communication between the TACACS+ server and client. By default, NVOS uses IP port 49.
- The TACACS timeout value, which is the number of seconds to wait for a response from the TACACS+ server before trying the next TACACS+ server. You can specify a value between 0 and 60. The default is 5 seconds.
- The TACACS+ authentication type. You can specify PAP to send clear text between the user and the server, CHAP to establish a PPP connection between the user and the server, or login. The default is PAP.

The following example commands set the timeout to 10 seconds and the TACACS+ server port to 32:

```
admin@nvos:~$ nv set system aaa tacacs timeout 10
admin@nvos:~$ nv set system aaa tacacs server SERVER1 port 32
admin@nvos:~$ nv config apply
```

The following example command sets the global authentication type to CHAP:

```
admin@nvos:~$ nv set system aaa tacacs authentication mode chap
admin@nvos:~$ nv config apply
```

5.3.5.6 TACACS+ Authorization

By default, TACACS+ performs authorization locally to ensure optimal performance.

However, it is possible to configure TACACS+ to perform authorization for every new connection using the nv set system aaa authorization mode command.

5.3.5.7 Troubleshooting

5.3.5.7.1 Show TACACS+ Configuration

Run the following commands to show TACACS+ configuration:

- To show all TACACS+ configuration (NVUE hides server secret keys), run the nv show system aaa tacacs command.
- To show TACACS+ authentication configuration, run the nv show system aaa tacacs authentication command.
- To show TACACS+ accounting configuration, run the nv show system aaa tacacs accounting command.
- To show TACACS+ server configuration, run the nv show system aaa tacacs server command.
- To show TACACS+ server priority configuration, run the nv show system aaa tacacs server <priority-id> command.
- To show the list of users excluded from TACACS+ server authentication, run the nv show system aaa tacacs exclude-user command.

The following example command shows all TACACS+ configuration:

```
admin@nvos:~$ nv show system aaa tacacs
                   applied
-----------------  -------
timeout            5
accounting
  state            enabled
authentication
  mode             pap
[server]           5
[server]           8
```

5.3.5.7.2 Incorrect Shared Key

The TACACS client on the switch and the TACACS server must have the same shared secret key. If this key is incorrect, the following message prints to `syslog`:

```
2017-09-05T19:57:00.356520+00:00 leaf01 sshd[3176]: nss_tacplus: TACACS+ server 192.168.0.254:49 read failed with
protocol error (incorrect shared secret?) user admin
```

5.3.5.7.3 Debug Issues with Accounting Records

If you add or delete TACACS+ servers from the configuration files, make sure you notify the `audisp` plugin with this command:

```
admin@nvos:~$ sudo killall -HUP audisp-tacplus
```

If accounting records do not send, add *debug=1* to the `/etc/audisp/audisp-tac_plus.conf` file, then run the command above to notify the plugin. Ask the TACACS+ user to run a command and examine the end of `/var/log/syslog` for messages from the plugin. You can also check the auditing log file `/var/log/audit audit.log` to be sure the auditing records exist. If the auditing records do not exist, restart the audit daemon with:

```
admin@nvos:~$ sudo systemctl restart auditd.service
```

5.3.5.7.4 TACACS+ Client Configuration Files

The following table describes the TACACS+ client configuration files that NVOS uses.

Filename	Description
`/etc/nsswitch.conf`	When the `libnss_tacplus` package installs, this file configures tacplus lookups through `libnss_tacplus`. If you replace this file by automation, you need to add tacplus as the first lookup method for the *passwd* database line.
`/etc/tacplus_nss.conf`	Sets the basic parameters for `libnss_tacplus`. The file includes a debug variable for debugging NSS lookups separately from other client packages.

Filename	Description
`/usr/share/pam-configs/tacplus`	The configuration file for `pam-auth-update` to generate the files in the next row. The file uses these configurations at `login`, by `su`, and by `ssh`.
`/etc/pam.d/common-*`	The `/etc/pam.d/common-*` files update for `tacplus` authentication. The files update with `pam-auth-update` when you install or remove `libpam-tacplus`.
`/etc/audit/audit.rules`	The audit rules file that generate when you install `auditd`.

5.3.5.8 TACACS Commands

- TACACS Commands

5.3.5.9 TACACS Commands

5.3.5.9.1 nv show system aaa tacacs accounting

	nv show system aaa tacacs accounting Show TACACS accounting configuration.
Syntax Description	N/A
Default	N/A
History	25.02.2002

Example	`admin@nvos:~$ nv show system aaa tacacs accounting`
	`operational applied`
	`----- ----------- --------`
	`state disabled disabled`

REST API	GET https://<ip>/nvue_v1/system/aaa/tacacs/accounting
Related Commands	nv set system aaa tacacs accounting state
Notes	Enable/disable tacacs accounting feature

5.3.5.9.2 nv show system aaa tacacs server

	nv show system aaa tacacs server Show remote TACACS servers.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command syntax
Example	admin@nvos:~$ nv show system aaa tacacs server Hostname Auth type Port Priority Secret Timeout ---------- --------- ---- -------- ------ ------- 10.2.30.40 pap 49 1 * 5
REST API	GET https://<ip>/nvue_v1/system/aaa/tacacs/server
Related Commands	nv set system aaa tacacs server
Notes	Show TACACS configured servers.

5.3.5.9.3 nv show system aaa tacacs authentication

	nv show system aaa tacacs authentication Show TACACS authentication global configuration.
Syntax Description	N/A
Default	N/A
History	25.02.3000
Example	admin@nvos:~$ nv show system aaa tacacs auth operational applied ---- ----------- ------- mode pap pap
REST API	GET https://<ip>/nvue_v1/system/aaa/tacacs/authentication
Related Commands	
Notes	

5.3.5.9.4 nv set system aaa tacacs accounting

	nv set system aaa tacacs accounting state <enabled | disabled> Configure remote TACACS servers.	
Syntax Description	state	enum: enabled | disabled
Default	N/A	

History	25.02.2002
Example	`admin@nvos:~$ nv set system aaa tacacs accounting state enabled`
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/accounting/state/<enabled/disabled>
Related Commands	nv show system aaa tacacs accounting nv show system aaa tacacs
Notes	

5.3.5.9.5 nv set system aaa tacacs server

	nv set system aaa tacacs server <server-id> Configure remote TACACS servers.	
Syntax Description	server-id	TACACS server ID: ipv4, ipv4-unicas, idn-server, ipv6
Default	N/A	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa tacacs server 1.2.3.4`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/server/<server-id>	
Related Commands	nv show system aaa tacacs server <server-id> nv show system aaa tacacs server nv show system aaa tacacs	
Notes		

5.3.5.9.6 nv set system aaa tacacs port

	nv set system aaa tacacs port <1–65535> Configure global TACACS authentication port.	
Syntax Description	port	Integer: 1–65535
Default	49	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa tacacs port 51`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/port	
Related Commands	nv show system aaa tacacs	
Notes		

5.3.5.9.7 nv set/unset system aaa tacacs authentication mode

	nv set system aaa tacacs authentication mode <pap | chap | login> nv unset system aaa tacacs authentication mode Configure global authentication type The unset form of the command sets the global authentication type to its default value.	
Syntax Description	auth-type	enum: chap, pap, login
Default	pap	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set sys aaa tacacs authentication mode chap`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/authentication	
Related Commands	nv show system aaa tacacs	
Notes		

5.3.5.9.8 nv set system aaa tacacs secret

	nv set system aaa tacacs secret <string | prompt> Configure global TACACS secret in cleartext.	
Syntax Description	secret	string | prompt
Default	""	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa tacacs secret tacacs-secret`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/secret	
Related Commands	nv show system aaa tacacs	
Notes		

5.3.5.9.9 nv set system aaa tacacs timeout

	nv set system aaa tacacs timeout <1–60> Configure the global tacacs reply timeout (seconds)	
Syntax Description	timeout	Integer: 1–60
Default	3	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system aaa tacacs timeout 5`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/timeout	

Related Commands	nv show system aaa tacacs
Notes	

5.3.5.9.10 nv set system aaa tacacs server auth-port

	nv set system aaa tacacs server <server-id> port <1–65535> Configure tacacs authentication port for server	
Syntax Description	server-id	TACACS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	auth-port	Integer: 1–65535
Default	None (set by TACACS global configuration)	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa taccs server 1.2.3.4 port 50`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/server/<server-id>/port/<port>	
Related Commands	nv show system aaa tacacs server <server-id> nv show system aaa tacacs server nv show system aaa tacacs	
Notes		

5.3.5.9.11 nv set system aaa tacacs server auth-mode

	nv set system aaa tacacs server <server-id> auth-mode <pap | chap | login> Configure TACACS authentication type for server.	
Syntax Description	server-id	TACACS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	auth-type	Enum: pap, chap, login
Default	None (set by tacacs global config)	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa tacacs server 1.2.3.4 auth-mode chap`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/server/<server-id>/auth-mode/<auth-mode>	
Related Commands	nv show system aaa tacacs server <server-id> nv show system aaa tacacs server nv show system aaa tacacs	
Notes		

5.3.5.9.12 nv set system aaa tacacs server secret

	nv set system aaa tacacs server <server-id> secret <string | prompt> Configure server secret in cleartext.

Syntax Description	server-id	TACACS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	secret	string | prompt
Default	None (set by tacacs global config)	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa tacacs server 1.2.3.4 secret Tacacs-Server-Password`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/server/<server-id>/password/<SECRET>	
Related Commands	nv show system aaa tacacs server <server-id> nv show system aaa tacacs server nv show system aaa tacacs	
Notes		

5.3.5.9.13 nv set system aaa tacacs server priority

	nv set system aaa tacacs server <server-id> priority <1–8> Configure tacacs priority for server.	
Syntax Description	server-id	Tacacs server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	priority	Integer: 1–8
Default	1	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa tacacs server 1.2.3.4 priority 5`	
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/server/<server-id>/priority/<priority-value>	
Related Commands	nv show system aaa tacacs server <server-id> nv show system aaa tacacs server nv show system aaa tacacs	
Notes		

5.3.5.9.14 nv set system aaa tacacs server timeout

	nv set system aaa tacacs server <server-id> timeout <1-60> Configure the reply timeout for a TACACS server (seconds).	
Syntax Description	server-id	TACACS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	timeout	Integer: 1–60
Default	None (set by TACACS global configuration)	
History	25.02.2002 25.02.3000 Updated command syntax	

Example	`admin@nvos:~$ nv set system aaa tacacs server 1.2.3.4 timeout 5`
REST API	SET https://<ip>/nvue_v1/system/aaa/tacacs/server/<server-id>/timeout/<timeout-value>
Related Commands	nv show system aaa tacacs server <server-id> nv show system aaa tacacs server nv show system aaa tacacs
Notes	

5.3.6 RADIUS

Various add-on packages enable RADIUS users to log in to NVOS switches in a transparent way with minimal configuration. There is no need to create accounts or directories on the switch. Authentication uses PAM and includes login, `ssh`, restapi, `sudo` and `su`.

5.3.6.1 RADIUS Client

RADIUS configuration is made of global configurations and per-server configurations. In general, if per-server configuration is not defined, the configuration will be taken from the global configuration.

All nv radius commands can be found in RADIUS Commands, where global ones are direct under /system/aaa/radius and per-server ones or under /system/aaa/radius/server/<hostname-id>

5.3.6.2 Radius Users

NVOS supports 3 types of RADIUS users, defined by Management-Privilege-Level configured in radius-server.

- Management-Privilege-Level := 15 # admin privileged users (nv set, nv config apply)
- Management-Privilege-Level := 7 # monitor privileged users (nv show)
- Management-Privilege-Level := 1 # non-privileged users (no nv commands access)

5.3.6.3 RADIUS Server Setup and Usage Example

Radius server can be configured either on a remote host, or on the switch itself (for testing or sanity-check).

5.3.6.3.1 Basic RADIUS Server Configuration

To conduct a basic RADIUS server configuration, add sections to "users" and "clients.conf" files.

User File Example

```
radius_user Cleartext-Password := "radius_user_password"
        Management-Privilege-Level := <15,7,1>
```

Client File Example

```
client client_name {
        ipaddr          = 10.1.2.3
        secret          = radius-secret
}
# Or as  CIDR block such as:
client 10.0.0.0/8 {
        secret          = testing-radius
}
```

5.3.6.3.2 How To Set Up Basic FreeRADIUS Server

1. Run the following command in a Debian machine or other similar Linux distributions.

```
sudo apt-get update
sudo apt-get install freeradius -y
```

2. Add your client IP to **/etc/freeradius/3.0/clients.conf** file as:

```
client client_name {
        ipaddr          = <CLIENT_IP>
        secret          = mysecret
}
```

or use CIDR block:

```
client 10.0.0.0/8 {
        secret          = global-secret
}
```

3. Add your required radius users to **/etc/freeradius/3.0/users** file as:

```
radius_admin_user Cleartext-Password := "radius_password"
        Management-Privilege-Level := 15

radius_monitor_user Cleartext-Password := "radius_password"
        Management-Privilege-Level := 7

radius_non_priv_user Cleartext-Password := "radius_password"
        Management-Privilege-Level := 1
```

4. Reboot freeRADIUS service (and make sure it is running).

```
sudo service freeradius restart
sudo service freeradius status
```

5. Configure RADIUS client to use such server.

```
admin@nvos:~$ nv set system aaa radius server <radius-server-ip> secret radius-secret
admin@nvos:~$ nv set system aaa authentication order radius local
admin@nvos:~$ nv config apply -y
```

Example:

```
admin@nvos:~$ nv set system aaa radius server 192.168.0.254 port 42
admin@nvos:~$ nv set system aaa radius server 192.168.0.254 secret 'myradius$key'
admin@nvos:~$ nv set system aaa radius server 192.168.0.254 priority 1
admin@nvos:~$ nv set system aaa authentication order radius local
admin@nvos:~$ nv config apply
```

6. Login with configured users.

5.3.6.4 Optional RADIUS Configuration

You can configure the following global RADIUS settings and server specific settings.

Setting	Description
retransmit	The maximum number of retransmission attempts allowed for requests when a RADIUS authentication request times out. This is a global option only; you cannot set the number of retransmission attempts for specific RADIUS servers.
timeout	The timeout value when a server is slow or latencies are high. You can set a value between 1 and 60. The default timeout is 3 seconds. If you configure multiple RADIUS servers, you can set a global timeout for all servers.
auth-type	RADIUS autherntication type to use.
statistics	Global confiugration to record RADIUS statistics.

The following example configures global RADIUS settings:

```
admin@nvos:~$ nv set system aaa radius retransmit 8
admin@nvos:~$ nv set system aaa radius timeout 10
admin@nvos:~$ nv set system aaa radius auth-type chap
admin@nvos:~$ nv set system aaa radius statistics enabled
admin@nvos:~$ nv config apply
```

The following example configures RADIUS settings for a specific RADIUS server:

```
admin@nvos:~$ nv set system aaa radius server 192.168.0.254 port 1811
admin@nvos:~$ nv set system aaa radius server 192.168.0.254 retransmit 5
admin@nvos:~$ nv set system aaa radius server 192.168.0.254 auth-type chap
admin@nvos:~$ nv set system aaa radius server 192.168.0.254 timeout 10
admin@nvos:~$ nv config apply
```

5.3.6.5 Show RADIUS Configuration

To show global RADIUS configuration, run the **nv show system aaa radius** command:

```
admin@nvos:~$ nv show system aaa radius
            operational  applied
----------  -----------  --------
port        1812         1812
auth-type   mschapv2     mschapv2
timeout     5            5
retransmit  0            0
statistics  disabled     disabled
[server]    192.168.0.254  192.168.0.254
```

To show all RADIUS configured servers, run the **nv show system aaa radius server** command:

```
admin@nvos:~$ nv show system aaa radius server
Server         Port  Priority  Password   Timeout
-------------  ----  --------  --------   -------
192.168.0.254  42    1         *          10
```

To show configuration for a specific RADIUS server, run the **nv show system aaa radius server <server>** command:

```
admin@nvos:~$ nv show system aaa radius server 192.168.0.254
          operational   applied
--------  ------------  ------------
port      42            42
timeout   10            10
secret    *             *
priority  1             1
```

5.3.6.6 Considerations

If two or more RADIUS users log in simultaneously, a UID lookup only returns the user that logs in first. Any process that either user runs applies to both, and all files that either user creates apply to the first name matched. This process is similar to adding two local users to the password file with the same UID and GID, and is an inherent limitation of using the UID for the fixed user from the password file. The current algorithm returns the first name matching the UID from the mapping file, which is either the first or second user that logs in.

5.3.6.7 RADIUS Commands

- RADIUS Commands

5.3.6.8 RADIUS Commands

5.3.6.8.1 nv show system aaa radius

	nv show system aaa radius Display RADIUS features configuration and state.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output

Example	<pre>admin@nvos:~$ nv show system aaa radius
	            operational applied
	---------- ----------- ----------
	port 1812 1812
	auth-type chap chap
	timeout 5 5
	secret * *
	retransmit 0 0
	statistics disabled disabled
	[server] 10.7.1.131</pre>

REST API	GET https://<ip>/nvue_v1/system/aaa/radius
Related Commands	
Notes	

5.3.6.8.2 nv show system aaa radius server

	nv show system aaa radius server Show remote RADIUS servers.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command syntax

Example	<pre>admin@nvos:~$ nv show system aaa radius server
	Hostname Auth port Auth type secret Priority Retransmit Timeout
	---------- --------- --------- -------- -------- ---------- -------
	1.1.1.1 1812 pap * 5 0 3
	10.7.34.20 1812 pap * 1 0 3</pre>

REST API	GET https://<ip>/nvue_v1/system/aaa/radius/hostname
Related Commands	nv show system aaa radius
Notes	

5.3.6.8.3 nv show system aaa radius server id

	nv show system aaa radius server<server-id> Display RADIUS server configuration.	
Syntax Description	server-id	RADIUS server ID: ipv4, ipv4-unicas, idn-hostname, ipv6
Default	N/A	
History	25.02.2002 25.02.3000 Updated command syntax	

Example	<pre>admin@nvos:~$ nv show system aaa radius server 10.7.34.20
	          operational applied
	-------- ----------- -------
	secret *
	priority 1</pre>	

REST API	GET https://<ip>/nvue_v1/system/aaa/radius/server/<server-id>	
Related Commands	nv show system aaa radius server nv show system aaa radius	

Notes	

5.3.6.8.4 nv set system aaa radius server

	nv set system aaa radius server <server-id> Configure remote RADIUS servers.	
Syntax Description	server-id	RADIUS server ID: ipv4, ipv4-unicas, idn-hostname, ipv6
Default	N/A	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	admin@nvos:~$ nv set system aaa radius server 1.2.3.4 operational applied -------- ----------- ------- secret * priority 1	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/server/<server-id>	
Related Commands	nv show system aaa radius server <server-id> nv show system aaa radius server nv show system aaa radius	
Notes		

5.3.6.8.5 nv set system aaa radius auth-port

	nv set system aaa radius auth-port <1–65535> Configure global RADIUS authentication port.	
Syntax Description	auth-port	Integer: 1–65535
Default	1812	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa radius auth-port 1813	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/auth-port	
Related Commands	nv show system aaa radius	
Notes		

5.3.6.8.6 nv set system aaa radius auth-type

	nv set system aaa radius auth-type <pap | chap> Configure global authentication type.	
Syntax Description	auth-type	enum: chap, pap
Default	pap	
History	25.02.2002	

Example	admin@nvos:~$ nv set system aaa radius auth-type chap
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/auth-type
Related Commands	nv show system aaa radius
Notes	

5.3.6.8.7 nv set system aaa radius retransmit

	nv set system aaa radius retransmit <0–10> Configure the default RADIUS retransmit tries.	
Syntax Description	retransmit	integer: 0–10
Default	0	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa radius retransmit 2	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/retransmit	
Related Commands	nv show system aaa radius	
Notes		

5.3.6.8.8 nv set system aaa radius secret

	nv set system aaa radius secret <string | prompt> Configure global radius server secret in cleartext.	
Syntax Description	secret	string, prompt
Default	""	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa radius secret Radius-secret	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/secret	
Related Commands	nv show system aaa radius	
Notes		

5.3.6.8.9 nv set system aaa radius statistics

	nv set system aaa radius statistics <enabled | disabled> Enable/disable log RADIUS statistics.	
Syntax Description	statistics	enum: enabled, disabled
Default	Disabled	
History	25.02.2002	

Example	admin@nvos:~$ nv set system aaa radius statistics enabled
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/statistics
Related Commands	nv show system aaa radius
Notes	

5.3.6.8.10 nv set system aaa radius timeout

	nv set system aaa radius timeout <1–60> Configure the global RADIUS server reply timeout (seconds).	
Syntax Description	timeout	Integer: 1–60
Default	3	
History	25.02.2002	
Example	admin@nvos:~$ nv set system aaa radius timeout 5	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/timeout	
Related Commands	nv show system aaa radius	
Notes		

5.3.6.8.11 nv set system aaa radius server auth-port

	nv set system aaa radius server <server-id> auth-port <1–65535> Configure RADIUS authentication port for server.	
Syntax Description	server-id	RADIUS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	auth-port	Integer: 1–65535
Default	None (set by RADIUS global config)	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	admin@nvos:~$ nv set system aaa radius server 1.2.3.4 auth-port 2812	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/server/<server-id>/auth-port/<auth-port>	
Related Commands	nv show system aaa radius server <server-id> nv show system aaa radius server nv show system aaa radius	
Notes		

5.3.6.8.12 nv set system aaa radius server auth-type

	nv set system aaa radius server <server-id> auth-type <pap | chap> Configure RADIUS authentication type for server.	
Syntax Description	server-id	RADIUS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	auth-type	Enum: pap, chap
Default	None (set by RADIUS global config)	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa radius server 1.2.3.4 auth-type chap`	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/server/<server-id>/auth-type/<auth-type>	
Related Commands	nv show system aaa radius server <server-id> nv show system aaa radius server nv show system aaa radius	
Notes		

5.3.6.8.13 nv set system aaa radius server secret

	nv set system aaa radius server <server-id> secret <secret> Configure server secret in cleartext.	
Syntax Description	server-id	RADIUS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	secret	string, prompt
Default	None (set by RADIUS global config)	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa radius server 1.2.3.4 secret Radius-Server-secret`	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/server/<server-id>/secret/<secret>	
Related Commands	nv show system aaa radius server <server-id> nv show system aaa radius server nv show system aaa radius	
Notes		

5.3.6.8.14 nv set system aaa radius server priority

	nv set system aaa radius server <server-id> priority <1–8> Configure RADIUS priority for server	
Syntax Description	server-id	RADIUS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	priority	Integer: 1–8

Default	1
History	25.02.2002 25.02.3000 Updated command syntax
Example	`admin@nvos:~$ nv set system aaa radius server 1.2.3.4 priority 5`
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/server/<server-id>/priority/<priority-value>
Related Commands	nv show system aaa radius server <server-id> nv show system aaa radius server nv show system aaa radius
Notes	

5.3.6.8.15 nv set system aaa radius server retransmit

	nv set system aaa radius server <server-id> retransmit <1–10> Configure the RADIUS retransmit tries for server.	
Syntax Description	server-id	RADIUS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	retransmit	Integer: 1–10
Default	None (set by RADIUS global config)	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa radius server 1.2.3.4 retransmit 5`	
REST API	SET https://<ip>/nvue_v1/system/aaa/radius/server/<server-id>/retransmit/<retransmit-value>	
Related Commands	nv show system aaa radius server <server-id> nv show system aaa radius server nv show system aaa radius	
Notes		

5.3.6.8.16 nv set system aaa radius server timeout

	nv set system aaa radius server <server-id> timeout <1–60> Configure the reply timeout for a RADIUS server (seconds).	
Syntax Description	server-id	RADIUS server ID: ipv4, ipv4-unicast, idn-hostname, ipv6
	timeout	Integer: 1–60
Default	None (set by RADIUS global config)	
History	25.02.2002 25.02.3000 Updated command syntax	
Example	`admin@nvos:~$ nv set system aaa radius server 1.2.3.4 timeout 5`	

REST API	SET https://<ip>/nvue_v1/system/aaa/radius/server/<server-id>/timeout/<timeout-value>
Related Commands	nv show system aaa radius server <server-id> nv show system aaa radius server nv show system aaa radius
Notes	

5.3.7 Role Based Access Control

The product features a Role-Based Access Control (RBAC) mechanism to manage user permissions effectively. This system ensures that users have access only to the functions necessary for their roles, enhancing both security and operational efficiency. The product supports two static roles: admin and monitor.

5.3.7.1 Default Roles

- The admin role has full access to all system configurations and management functionalities.
- The monitor role provides read-only access, allowing users to view system information and monitor operations without making any changes.

5.3.7.2 Custom Roles

Users can define new accounts with one of the available roles and modify the role of existing users as needed.

```
admin@nvos:~$ nv set system aaa user new_monitor role monitor
```

The roles of the default accounts (admin and monitor) cannot be changed

In addition to the default roles that NVOS provides, you can create your own roles to restrict authorization, giving you more granular control over what a user can manage on the switch. For example, you can assign a user the role of network manager and provide the user privileges for

interface management, service management and system management. When the user logs in and executes an NVUE command, NVUE checks the user privileges and authorizes the user to run that command.

5.3.7.2.1 Elements of Custom Role-Based Access Control

Custom role-based access control consists of the following elements:

Element	Description
Role	A virtual identifier for multiple classes (groups). You can assign only one role for a user. For example, for a user that can manage interfaces, you can create a role called `IBMgr`.
Class	A class is similar in concept to a Linux group. Creating and managing classes is the simplest way to configure multiple users simultaneously, especially when configuring permissions. A class consists of: • Command paths, which NVOS bases on the objects in the NVUE declarative model and, which are the same as URI paths; for example; you can use the `/interface/` command path to allow or deny a user access to all interfaces, or `/system/ntp` to allow or deny a user access to ntp configuration. Use the tab key to see available command paths (`nv set system aaa class <class-name> command-path / <<press tab>>`). • Permissions for the command paths: (`ro`) to run show commands, (`rw`) to run set, unset, and apply commands, (`act`) to run action commands, or (`all`) to run all commands. The default permission setting is `all`.
Action	The action for the class: `allow` or `deny`.

- You can assign a maximum of 64 classes to a role.
- You can configure a maximum of 128 command paths for a class.
- When you configure a command path, you allow or deny a specific schema path and its children. For example the command path `/interface/` allows or denies access to all interface commands, whereas the command path `/interface/eth0` allows or denies access to eth0 commands.

5.3.7.2.2 Example Role Permissions

The following example describes the permissions for a role (`role1`) that consists of three classes: `class1`, `class2`, `class3`.

class1 has the `allow` class action and the following command path permissions:

Command Path	Permissions
`/interface/`	`all`
`/ib/`	ro

class2 has the `allow` class action and the following command path permissions:

Command Path	Permissions
`/system/`	`ro`
`/ib/`	`act`
/vrf/	rw

class3 has the `deny` class action and the following command path permissions:

Command Path	Permissions
`/interface/eth0/`	ro

The following table shows the permissions for a user assigned the role `role1`. In the table, R is read only (RO), W is write, and X is action (ACT).

Path	Allow	Deny	Permissions
`/platform/`		RWX	Implicit deny
`/acl/`		RWX	Implicit deny
All unspecified paths are implicit deny			
`/interface/`	RWX		The permissions specified
`/interface/eth0`		RWX	The permissions specified
All unspecified children of `/interface/` inherit parent permissions	RWX		
All unspecified children of `/interface/eth0` inherit parent permissions		RWX	
`/system/`	R	WX	The permissions specified
`/system/aaa/`	R	WX	Inherited from parent
`/system/cpu/`	R	WX	Inherited from parent
All unspecified children of `/system/` inherit parent permissions	R	WX	
/ib/	RX	W	The permissions specified
/ib/ibdiagnet	RX	W	Inherited from parent
`/vrf/`	RW	X	The permissions specified
All unspecified children of `/vrf/` inherit parent permissions	RW	X	

5.3.7.3 Assign a Custom Role to a User Account

To assign a custom role to a user account:

1. Create a role and classes for the role.

2. Assign the action (allow or deny) for each class.
3. Add command paths and permissions for each class.
4. Assign a role to a user.

You assign a custom role to an existing user account.

The following example creates and assigns the three classes described above for role `role1`.

`class1` has permissions to manage all interfaces and to show all the tree of ib:

```
admin@nvos:~$ nv set system aaa role role1 class class1
admin@nvos:~$ nv set system aaa class class1 action allow
admin@nvos:~$ nv set system aaa class class1 command-path /interface/ permission all
admin@nvos:~$ nv set system aaa class class1 command-path /ib/ permission ro
admin@nvos:~$ nv config apply
```

`class2` has permissions to show system commands, perform actions on ib cmd and show and set configuration of vrf:

```
admin@nvos:~$ nv set system aaa role role1 class class2
admin@nvos:~$ nv set system aaa class class2 command-path /system/ permission ro
admin@nvos:~$ nv set system aaa class class2 command-path /ib/ permission act
admin@nvos:~$ nv set system aaa class class2 command-path /vrf/ permission rw
admin@nvos:~$ nv config apply
```

`class3` denies permission of `ro` on eth0 interface, this explicit definition on the child causes it to stop inheriting the parent read and action permissions.

```
admin@nvos:~$ nv set system aaa class class3 action deny
admin@nvos:~$ nv set system aaa class class3 command-path /interface/eth0/ permission ro
admin@nvos:~$ nv set system aaa role role1 class class3
admin@nvos:~$ nv config apply
```

The following command assigns user `ib_user` the role `role1`:

```
admin@nvos:~$ nv set system aaa user ib_user role role1
admin@nvos:~$ nv config apply
```

5.3.7.4 Delete Custom Roles

To delete a custom role and all its classes, you must first unassign the role from the user, then delete the role:

```
admin@nvos:~$ nv unset system aaa user ib_user role role1
admin@nvos:~$ nv unset system aaa role role1
admin@nvos:~$ nv config apply
```

To delete a class from a role, run the `nv unset system aaa role <role> class <class>` command:

```
admin@nvos:~$ nv unset system aaa role role1 class class2
```

```
admin@nvos:~$ nv config apply
```

5.3.7.5 Show Custom Role Information

To show the user accounts configured on the system, run the NVUE **nv show system aaa user** command.

```
admin@nvos:~$ nv show system aaa user
Username  Full-name             Role     state
--------  --------------------  -------  -------
admin     System Administrator  admin    enabled
ib_user                         role1    enabled
monitor   System Monitor        monitor  enabled
```

To show information about a specific user account including the role assigned to the user, run the NVUE **nv show system aaa user <user>** command:

```
admin@nvos:~$ nv show system aaa user ib_user
                    operational  applied
------------------  -----------  -------
state               enabled      enabled
role                role1        role1
full-name
password            *            *
hashed-password     *
ssh
  [authorized-key]
```

To show all the roles configured on the switch, run the NVUE **nv show system aaa role** command:

```
admin@nvos:~$ nv show system aaa role
Role Name  Class
---------  --------
admin      nvaction
           nvapply
           sudo
monitor    nvshow
role1      class1
           class2
           class3
```

To show the classes applied to specific role, run the **nv show system aaa role <role>** command:

```
admin@nvos:~$ nv show system aaa role role1
        operational             applied
-------  ----------------------  -------
groups   adm,class1,class2,class3
[class]  class1                  class1
[class]  class2                  class2
[class]  class3                  class3
```

To show all the classes configured on the switch, run the **nv show system aaa class** command:

```
admin@nvos:~$ nv show sys aaa class
Class Name  Command Path      Permission  Action
----------  ----------------  ----------  ------
class1      /ib/              ro          allow
            /interface/       all
class2      /ib/              act         allow
            /system/          ro
            /vrf/             rw
class3      /interface/eth0/  all         deny
```

```
nvaction    /               act         allow
nvapply     /               rw          allow
nvshow      /               ro          allow
sudo        /               all         allow
```

To show the configuration and state of the command paths for a class, run the **nv show system aaa class <class>** command:

```
admin@nvos:~$ nv show system aaa class class3
                applied
--------------  ----------------
action          deny
[command-path]  /interface/eth0/
```

5.3.7.6 Role Based Access Control Commands

- Role Based Access Control Commands

5.3.7.7 Role Based Access Control Commands

5.3.7.7.1 nv show system aaa role

	nv show system aaa role Displays list of roles (user capabilities) and their groups.
Syntax Description	N/A
Default	N/A
History	25.02.3000
Example	 admin@nvos:~$ nv show system aaa role Role Name Class --------- -------- admin nvaction nvapply sudo monitor nvshow
REST API	GET https://<ip>/nvue_v1/system/aaa/role

Related commands	nv show system aaa role monitor nv show system aaa role <role-id> nv set system aaa role <role-id> class <class-id>
Notes	• admin—full administrative capabilities • monitor—read only capabilities, can not change the running configuration

5.3.7.7.2 nv show system aaa role id

	nv show system aaa role <role-id> Displays configuration of a role.	
Syntax Description	role-id	The name of the role (i.e., admin, monitor)
Default	N/A	
History	25.02.3000	
Example	admin@nvos:~$ nv show system aaa role role1 admin@nvos:~$ nv show system aaa role monitor operational applied ------- ----------- ------- groups adm,nvshow [class] nvshow nvshow	
REST API	GET https://<ip>/nvue_v1/system/aaa/role/{role-id}	
Related commands	nv show system aaa role nv set system aaa role <role-id> class <class-id>	
Notes	admin—full administrative capabilities monitor—read only capabilities, cannot change the running configuration	

5.3.7.7.3 nv show system aaa class

	nv show system aaa class Display all Classes configuration and state.	
Syntax Description	N/A	
Default	N/A	
History	25.02.3000	
Example	admin@nvos:~$ nv show system aaa class Class Name Command Path Permission Action ---------- ------------ ---------- ------ nvaction / act allow nvapply / rw allow nvshow / ro allow sudo / all allow	
REST API	GET https://<ip>/nvue_v1/system/aaa/class	
Related Commands	nv set system aaa class <class-id> command-path <command-path-id> nv set system aaa class <class-id> action <arg>	
Notes		

5.3.7.7.4 nv show system aaa class id

	nv show system aaa class <class-id> Display configuration and state of a class.	
Syntax description	class-id	The name of the class.
Default	N/A	
History	25.02.3000	
Example	admin@nvos:~$ nv show sys aaa class nvshow applied -------------- ------- action allow [command-path] /	
REST API	GET https://<ip>/nvue_v1/system/aaa/class/{class-id}	
Related Commands	nv set system aaa class <class-id> command-path <command-path-id> nv set system aaa class <class-id> action <arg> nv show system aaa class	
Notes		

5.3.7.7.5 nv show system aaa class id command-path

	nv show system aaa class <class-id> command-path [<command-path-id>] Display configuration and state of a class command-paths.	
Syntax description	class-id	The name of the class
	command-path-id	The command path (e.g., /interface/eth0)
Default	N/A	
History	25.02.3000	
Example	admin@nvos:~$ nv show sys aaa class nvshow command-path Command Path Permission ------------ ---------- / ro	
REST API	GET https://<ip>/nvue_v1/system/aaa/class/{class-id}/command-path/{command-path-id}	
Related Commands	nv set system aaa class <class-id> command-path <command-path-id> nv set system aaa class <class-id> action <arg> nv show system aaa class	
Notes		

5.3.7.7.6 nv set/unset system aaa class action

	nv set system aaa class <class-id> action <arg> nv unset system aaa class <class-id> action Set the action to be taken upon getting a match on the command paths. Unset the action to be taken upon getting a match on the command paths.

Syntax Description	class-id	The name of the class
	arg	The action to be taken upon getting a match on the command paths enum: allow, deny
Default	action: allow	
History	25.02.3000	
Example	admin@nvos:~$ nv set system aaa **class** ib_enjoyer action allow	
REST API	N/A	
Related Commands		
Notes		

5.3.7.7.7 nv set/unset system aaa class command-path

	nv set system aaa class <class-id> command-path [<command-path-id>] [permission <permission>] nv unset system aaa class <class-id> command-path [<command-path-id>] [permission] Configure command paths for classes. The unset form of the command clears command paths under classes.	
Syntax Description	class-id	The name of the class
	command-path-id	The command path (e.g., /interface/eth0)
	permission	The permissions on the command path enum: ro, rw, act, all
Default	permission: all	
History	25.02.3000	
Example	admin@nvos:~$ nv set system aaa **class** class3 command-path /**interface**/eth0/ permission all	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/class/{class-id}/command-path/{command-path-id}	
Related Commands	nv set system aaa class <class-id> action <arg>	
Notes		

5.3.7.7.8 nv unset system aaa class

	nv unset system aaa class [<class-id>] Clear class configuration.	
Syntax Description	class-id	The name of the class
Default	N/A	
History	25.02.3000	

Example	admin@nvos:~$ nv unset system aaa class ib_enjoyer
REST API	PATCH https://<ip>/nvue_v1/system/aaa/class/{class-id}
Related Commands	nv set system aaa class <class-id> command-path <command-path-id> nv set system aaa class <class-id> action
Notes	A class cannot be unset if it is a part of a role.

5.3.7.7.9 nv set/unset system aaa role class

	nv set system aaa role {<role-id> class <class-id>} nv unset system aaa role {<role-id> class [<class-id>]} Configure classes under role. The unset form of the command clears classes under role.	
Syntax Description	role-id	The name of the role
	class-id	The name of the class
Default	N/A	
History	25.02.3000	
Example	admin@nvos:~$ nv set system aaa role role1 class class3	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/role/{role-id}/class/{class-id}	
Related Commands	nv set system aaa class <class-id> action <arg> nv set system aaa class <class-id> command-path <command-path-id> nv set/unset system aaa user role	
Notes		

5.3.7.7.10 nv unset system aaa role

	nv unset system aaa role <role-id> Clear role's configuration.	
Syntax Description	role-id	The name of the role
Default	N/A	
History	25.02.3000	
Example	admin@nvos:~$ nv unset system aaa role role1	
REST API	PATCH https://<ip>/nvue_v1/system/aaa/role/{role-id}/	
Related Commands	nv set system aaa role <role-id> class <class-id>	
Notes	A role cannot be unset if it is a part of a user.	

5.3.8 Authorization

Authorization defines the capabilities, privileges, and available commands for a specific user within the system.

Remote user authorization defines the user's permission level through server configuration, as specified by the remote AAA server.

The Authorization section supports configuring when user authorization is derived from the AAA server.

> ⚠ Currently, remote authorization is supported only when TACACS is used.

```
admin@nvos:~$ nv set sys aaa authorization mode session-remote
# or
admin@nvos:~$ nv set sys aaa authorization mode first-login
```

Default mode: first-login

- session-remote—The user's permission level is validated and updated from the AAA server each time they log in (per session).
- first-login—The user's permission level is retrieved from the AAA server only during the first login and is then cached for subsequent logins.

5.3.8.1 Authorization Commands

- Authorization Commands

5.3.8.2 Authorization Commands

5.3.8.2.1 nv show system aaa authorization

	nv show system aaa tacacs authorization Show authorization configuration.
Syntax Description	N/A
Default	N/A
History	25.02.5030

Example	`admin@nvos:~$ nv show system aaa authorization`
	`operational applied`
	`---- ------------ ------------`
	`mode session-remote session-remote`

REST API	GET https://<ip>/nvue_v1/system/aaa/authorization
Related Commands	nv set system aaa authorization mode
Notes	This configuration applies only when TACACS authentication is configured. It does not affect RADIUS or LDAP.

5.3.8.2.2 nv set system aaa authorization mode

	nv set system aaa authorization mode <authorization-mode> Sets system authorization mode.	
Syntax Description	authorization-mode	enum: first-login-remote, session-remote
Default	first-login-remote	
History	25.02.5030	
Example	admin@nvos:~$ nv set system aaa authorization mode	
REST API	SET https://<ip>/nvue_v1/system/aaa/authorization/mode	
Related commands	nv show sys aaa authorization	
Notes	This configuration applies only when TACACS authentication is configured. It does not affect RADIUS or LDAP.	

5.3.9 SSH for Remote Access

5.3.9.1 Authorized SSH Key

To install an authorized SSH key, add the contents of a user's SSH public key to the authorized keys file (**~/.ssh/authorized_keys**) for that user.

An SSH public key is a text string composed of three space-separated fields:

```
<type> <key string> <comment>
```

5.3.9.1.1 Installing an Authorized SSH Key

Use the following NVUE commands to configure an authorized SSH key for a user:

```
admin@nvos:~$ nv set system aaa user admin2 ssh authorized-key prod_key key XABDB3NzaC1yc2EAAAADAQABAAABgQCvjs/
RFPhxLQMkckONg+1RE1PTIO2JQhzFN9TRg7ox7o0tfZ+IzSB99lr2dmmVe8FRWgxVjc...
admin@nvos:~$ nv set system aaa user admin2 ssh authorized-key prod_key type ssh-rsa
admin@nvos:~$ nv config apply
```

5.3.9.1.2 Public Key Authentication (PKA)

Public Key Authentication (PKA), also known as SSH key authentication, uses a public–private key pair generated by a key generation tool to authenticate a user to the SSH server.

- The private key remains on the SSH client and is typically password-protected.
- The public key is stored on the SSH server.

5.3.9.1.2.1 Enforcing PKA-Only Authentication

To allow only key-based authentication and disable password authentication for users with private keys, enable PKA-only mode:

```
admin@nvos:~$ nv set system ssh-server pka-only enabled
admin@nvos:~$ nv config apply
```

5.3.9.2 Certificate-Based Authentication

As an alternative to passwords and individual SSH keys, you can enable certificate-based authentication. This method uses a trusted Certificate Authority (CA) to authenticate users, automatically enforces certificate expiration, and eliminates Trust-On-First-Use (TOFU) risks.

5.3.9.2.1 Configuring Certificate-Based Authentication

To configure certificate-based authentication for a user:

1. Set the trusted CA key ID, literal, and type. You can see the key ID, literal, and type in a public key file.
2. Enable certificate authentication for the user.
3. (Optional) Assign one or more certificate principals to the user. If no principal is specified, the user is treated as the sole principal.

5.3.9.2.2 Example Configuration

The following example sets the trusted CA key (`KEY1`) as type `ssh-rsa` with the specified literal (`AAAAB3NzaC1yc2EAAAADA..`), enables certificate authentication for the user `ADMIN1` ,and adds the principle `aaa` :

```
admin@nvos:~$ nv set system ssh-server trusted-ca-keys KEY1 key AAAAB3NzaC1yc2EAAAADA..
admin@nvos:~$ nv set system ssh-server trusted-ca-keys KEY1 type ssh-rsa
admin@nvos:~$ nv set system aaa user ADMIN1 ssh cert-auth state enabled
admin@nvos:~$ nv set system aaa user ADMIN1 ssh cert-auth principals aaa
admin@nvos:~$ nv config apply
```

5.3.9.2.3 Viewing Trusted CA Configuration

To display the trusted CA certificate authentication configuration, run:

```
admin@nvos:~$ nv show system ssh-server trusted-ca-keys
      operational  applied
---- ------------ --------
```

```
key    *            *
type   ssh-rsa      ssh-rsa
```

5.3.9.3 SSH for Remote Access Commands

- SSH for Remote Access Commands

5.3.9.4 SSH for Remote Access Commands

5.3.9.4.1 nv show system ssh-server trusted-ca-keys

	nv show system ssh-server trusted-ca-keys Display ssh trusted ca keys	
Syntax Description	N/A	
Default	N/A	
History	25.02.7002	
Example	admin@nvos:~$nv show sys ssh-server trusted-ca-keys Trusted CA Key Name Key string Key Type ------------------ ---------- -------- k1 * ssh-rsa	
REST API	GET https://<ip>/nvue_v1//system/ssh-server/trusted-ca-keys	
Related Commands	nv show sys ssh-server trusted-ca-keys	
Notes		

5.3.9.4.2 nv show system ssh-server trusted-ca-keys

	nv show system ssh-server trusted-ca-keys Display configuration of one ssh trusted ca key	
Syntax Description	key-id	Id used during ca-key set
Default	N/A	
History	25.02.7002	

Example	admin@nvos:~$nv show sys ssh-server trusted-ca-keys k1 operational applied ---- ----------- ------- key * type ssh-rsa
REST API	GET https://<ip>/nvue_v1/system/ssh-server/trusted-ca-keys/{key_id}
Related Commands	nv set sys ssh-server trusted-ca-keys {key_id} key {key_valie} nv show sys ssh-server trusted-ca-keys
Notes	

5.3.9.4.3 nv set system ssh-server trusted-ca-keys key value

	nv set system ssh-server trusted-ca-keys {key_id} key {key_value} Display ssh trusted ca keys	
Syntax Description	key_id	ID of key to set.
	key_value	Encoded trusted CA key value.
Default	N/A	
History	25.02.7002	
Example	admin@nvos:~$nv set system ssh-server trusted-ca-keys k1 key AAAAB3.....	
REST API	GET https://<ip>/nvue_v1//system/ssh-server/trusted-ca-keys/{key_id}	
Related Commands	nv show sys ssh-server trusted-ca-keys nv set system ssh-server trusted-ca-keys k1 type	
Notes		

5.3.9.4.4 nv set system ssh-server trusted-ca-keys type key type

	nv set system ssh-server trusted-ca-keys {key_id} type {key_type} Display ssh trusted ca keys	
Syntax Description	key_id	ID of key to set.
	key_type	ssh-rsa | ecdsa-sha2-nistp256 | ecdsa-sha2-nistp384 | ecdsa-sha2-nistp521 | ssh-ed25519
Default	key_type : ssh-rsa	
History	25.02.7002	
Example	admin@nvos:~$nv set system ssh-server trusted-ca-keys k1 type ssh-rsa	
REST API	GET https://<ip>/nvue_v1//system/ssh-server/trusted-ca-keys/{key_id}	
Related Commands	nv show sys ssh-server trusted-ca-keys nv set system ssh-server trusted-ca-keys k1 type	
Notes		

5.3.9.4.5 nv show system aaa user ssh cert-auth principals

	nv show system aaa user {user} ssh cert-auth principals Display user principals, to be used for trusted ca connection	
Syntax Description	N/A	
Default	N/A	
History	25.02.7002	
Example	admin@nvos:~$nv show sys aa user test ssh cert-auth principals Certificate Principal --------------------- admin admin2	
REST API	GET https://<ip>/nvue_v1//system/aaa/user/test/ssh/cert-auth/principals	
Related Commands	nv set system aaa user {user} ssh cert-auth principals	
Notes		

5.3.9.4.6 nv show system aaa user {user} ssh cert-auth

	nv show system aaa user {user} ssh cert-auth / Display user certificate authentication information	
Syntax Description	N/A	
Default	N/A	
History	25.02.7002	
Example	admin@nvos:~$nv show sys aa user test ssh cert-auth operational applied pending ---------- ---------- ------- ------- state enabled enabled enabled [principals] admin admin admin	
REST API	GET https://<ip>/nvue_v1//system/aaa/user/test/ssh/cert-auth	
Related Commands	nv set system aaa user {user} ssh cert-auth principals	
Notes		

5.3.9.4.7 nv set system aaa user ssh cert-auth principals

	nv set system aaa user {user} ssh cert-auth principals {principal} Set SSH trusted CA principal for user.	
Syntax Description	user	User ID.
	principal	Principal string, 1 item per command.
Default	N/A	
History	25.02.7002	

Example	 admin@nvos:~$nv set sys aaa user admin ssh cert-auth principals test	
REST API	GET https://<ip>/nvue_v1//system/aaa/user/test/ssh/cert-auth	
Related Commands	nv show system aaa user {user} ssh cert-auth principals	
Notes		

5.3.9.4.8 nv set system aaa user ssh cert-auth state

	nv set system aaa user {user} ssh cert-auth state {enabled|disabled} Enable/Disable SSH trusted CA principal for user.	
Syntax Description	user	User ID.
	state	enabled | disabled
Default	N/A	
History	25.02.7002	
Example	 admin@nvos:~$nv set sys aaa user test ssh cert-auth state enabled	
REST API	GET https://<ip>/nvue_v1//system/aaa/user/test/ssh/cert-auth	
Related Commands	nv show system aaa user {user} ssh cert-auth principals	
Notes		

5.4 Certificates Management

NVOS includes a self-signed certificate and private key to use on the server so that it works out of the box. The switch generates the self-signed certificate and private key when it boots for the first time. The X.509 certificate with the public key is in `/etc/ssl/certs/nvue.pem` and the corresponding private key is in `/etc/ssl/private/nvue.key`.

NVIDIA recommends you use your own certificates and keys.

NVOS lets you manage CA certificates (such as DigiCert or Verisign) and entity (end-point) certificates. Both a CA certificate and an entity certificate can contain a chain of certificates. The CA certificates can be also addressed as trust bundles, which means that CA certs can also include intermediate certificates.

You can import certificates onto the switch (fetch certificates from an external source), set which certificate you want to use for the NVUE REST API, gNMI, NMX, and show information about a certificate, such as the serial number, and the date and time during which the certificate is valid.

NVOS uses OpenSSL 3, which does not support legacy ciphers by default. Ensure all certificates and keys use modern algorithms (e.g., SHA-256, RSA, ECDSA) and avoid deprecated ones such as MD5, SHA1, RC2-40-CBC, RC4, and DES.

5.4.1 Import Certificate

- A maximum of 25 entity certificates and a maximum of 25 CA certificates can be imported
  - A single CA certificate entry may contain up to 100 PEM strings and may also include intermediate certificates.
- The imported server/entity certificate contains sensitive private key information. NVIDIA recommends that you use a secure transport such as SFTP, SCP, or HTTPS.
- To import an entity certificate, run an nv action import system security certificate <cert-id> command.
- To import a CA certificate, run an nv action import system security ca-certificate <cert-id> command.

If the certificate is passphrase protected, the passphrase to be included.

You must provide a certificate ID (<cert-id>) to uniquely identify the certificate you import.

5.4.1.1 CA Certificate

The following example imports a CA certificate with a public key and calls the certificate tls-cert-1. The public key is a Base64 ASCII encoded PEM string.

```
nvos@switch:~$ nv action import system security ca-certificate tls-cert-1 data """-----BEGIN CERTIFICATE----- TODO
----END CERTIFICATE-----"""
```

The following example imports a CA certificate with a public key and calls the certificate `tls-cert-1` with URI `scp://user@pass:1.2.3.4:/opt/certs/ca-cert.crt`.

```
nvos@switch:~$ nv action import system security ca-certificate tls-cert-1 uri scp://user@pass:1.2.3.4:/opt/certs/
ca-cert.crt
```

The following example imports a CA certificate with a public key and calls the certificate `tls-cert-1` with URI `scp://user@pass:1.2.3.4:/opt/certs/ca-cert.crt`. External makes CA certificate standalone, which means certificate is not attached to common system CA certificates bundle in `/etc/ssl/certs/ca-certificates.crt`.

```
nvos@switch:~$ nv action import system security ca-certificate tls-cert-1 uri scp://user@pass:1.2.3.4:/opt/certs/
ca-cert.crt external
```

5.4.1.2 Import Certificate

The following example imports an entity certificate bundle (public + private key) and calls the certificate tls-cert-1. The certificate bundle is passphrase protected with mypassphrase.

A certificate bundle must be in .p12 format.

To generate a certificate bundle with the .p12 format, use the following command:

```
openssl pkcs12 -export -out $cert_filename.p12 -in $cert_filename.pem -inkey $cert_filename.key -passout pass:
$p12_pass
```

Example of importing the.p12 boundle file:

```
nvos@switch:~$ nv action import system security certificate tls-cert-1 passphrase mypassphrase uri-bundle scp://
user@pass:1.2.3.4:/opt/certs/cert.p12
```

The following example imports an entity certificate bundle and calls the certificate tls-cert-local. The certificate is located on top of the local machine under `/home/admin` path:

```
nv action import system security certificate tls-cert-local uri-public-key file://127.0.0.1/home/admin/cert.crt
uri-private-key file://127.0.0.1/home/admin/cert.key

#Using absolute path is also possible. Example:
nv action import system security certificate tls-cert-local uri-public-key file:///home/admin/cert.crt uri-private-
key file:///home/admin/cert.key
```

The following example imports an entity certificate with the public key URI `scp://user@pass:1.2.3.4` and private key URI `scp://user@pass:1.2.3.4`, and calls the certificate `tls-cert-1`. The certificate is not passphrase protected.

```
nvos@switch:~$ nv action import system security certificate tls-cert-1 uri-public-key scp://user@pass:1.2.3.4 uri-
private-key scp://user@pass:1.2.3.4
```

5.4.2 Set the Certificate to Use in NVUE REST API

> ⚠ When updating a new CA/Certificate/CRL, make sure to perform the "nv config save" before proceeding with a reboot.

You can configure the NVUE REST API to use a specific certificate.

The following example configures the API to use the certificate tls-cert-1:

```
nvos@switch:~$ nv set system api certificate tls-cert-1

nvos@switch:~$ nv config apply
```

The following example configures the API to use the self-signed certificate:

```
nvos@switch:~$ nv set system api certificate self-signed
```

```
nvos@switch:~$ nv config apply
```

To unset the certificate to use with the NVUE REST API:

```
nvos@switch:~$ nv unset system api certificate tls-cert-1
```

5.4.3 Set the Certificate to Use for GNMI Service

You can configure the GNMI to use a specific certificate.

The following example configures the API to use the certificate tls-cert-1:

```
nvos@switch:~$ nv set system gnmi-server certificate tls-cert-1
nvos@switch:~$ nv config apply
```

The following example configures the API to use the self-signed certificate:

```
nvos@switch:~$ nv set system gnmi-server certificate self-signed
nvos@switch:~$ nv config apply
```

To unset the certificate to use with the NVUE REST API:

```
nvos@switch:~$ nv unset system gnmi-server certificate tls-cert-1
```

5.4.4 Delete Certificates

- To delete an entity certificate and the key data stored on the switch, run the nv action delete system security certificate <cert-id> command.
- To delete a CA certificate and the key data stored on the switch, run the nv action delete system security ca-certificate <cert-id> command.

The following command deletes the certificate tls-cert-1:

```
nvos@switch:~$ nv action delete system security certificate tls-cert-1
```

5.4.5 Show Certificate Information

- To show all the entity certificates on the switch, run the nv show system security certificate command.
- To show all the CA certificates on the switch, run the nv show system security ca-certificate command.

The following example shows all the entity certificates on the switch:

```
nvos@switch:~$ nv show system security certificate
```

- To show the applications that are using a specific entity certificate, run the nv show system security certificate <cert-id> installed command.

- To show the applications that are using a specific CA certificate, run the nv show system security ca-certificate <cert-id> installed command.

The following example shows the applications that are using a specific entity certificate.

```
nvos@switch:~$ nv show system security certificate tls-cert-1 installed
```

- To show detailed information about a specific entity certificate, run the nv show system security certificate <cert-id> dump command.
- To show detailed information about a specific CA certificate, run the nv show system security ca-certificate <cert-id> dump command.

The following example shows detailed information about the CA certificate tls-cert-1:

```
nvos@switch:~$ nv show system security ca-certificate tls-cert-1 dump
```

5.4.6 Certificate Revoke List (CRL)

With mTLS feature, NVUE Server {Nginx} uses CA certificates from trusted store to validate the client certificate used in request. These CAs also issue Certificate Revocation List (CRL) from time to time to revoke certificates before their expiration dates. Reason for revocation are:

- Certificate Compromise: The private key associated with a certificate has been compromised or leaked.
- CA Compromise: The Certificate Authority's private key is compromised.
- Superseded Certificate A newer certificate replaces an existing certificate due to upgrades or changes in cryptographic algorithms, key lengths, or certificate details.
- Affiliation Change The organization, individual, or domain associated with the certificate changes ownership or no longer exists.
- Invalid Information The certificate contains incorrect or outdated information, such as an invalid domain name, organization details, or expiration date.

Each CRL includes

- CRL Number, Last Update and Next Update.
- CRL number: sequentially increasing CRL number.
- Last Update: issued date of current (this) CRL.
- Next Update: when the new CRL should be issued, also expiration date for current CRL.
- A list of revoked certificates:
- Serial number.
- Date of revocation.
- The reason for revocation (optional, depending on CA policy).

On Nvidia Switch, CRL is applicable

- mTLS when an application runs as server.
- CRL is used while validating client certificates.
- TLS when an application runs as client.
- CRL is used while validating peer server certificates.
- supported CRL format PEM
- Example feature that using CRL: NMX, GNMI, NGINX

5.4.6.1 Import CRL

The following command is an example of how to import a CRL file to the device. (similar to certificates import)

```
# local import of CRL
nv action import system security crl crl_1 uri file:///tmp/crl3/demoCA/crl/client_ca.crl
# remote import of CRL
nv action import system security crl crl_1 uri scp://user@pass:1.2.3.4:/tmp/crl3/demoCA/crl/client_ca.crl
```

5.4.7 Certificate Management Commands

- Certificate Management Commands

5.4.8 Certificate Management Commands

5.4.8.1 nv show system security ca-certificate

	nv show system security ca-certificate Display owned CA certificates.
Syntax Description	N/A
Default	N/A

History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system security ca-certificate Certificate ID Serial Number Valid From Valid To Summary ca_id_168 43:69:66:63:11:75:31:00:95:03:96: 2025-02-18T16:06:37+02:00 2026-02-18T16:06:37+02:00 count: 1 Installed: nvue-rest-api
REST API	GET https://<ip>/nvue_v1/system/security/ca-certificate/
Related Commands	nv action import system security ca-certificate {cacert id}
Notes	

5.4.8.2 nv show system security certificate

	nv show system security certificate Display owned certificates.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system security certificate Certificate ID Serial Number Valid From Valid To Summary cert_id_396 5A:A6:4C:4A:8C:B8:9A:89:8 2025-02-18T16:06:37+02:00 2026-02-18T16:06:37+02:00 Installed: nvue-rest-api Installed: gnmi-server
REST API	GET https://<ip>/nvue_v1/system/security/ca-certificate/
Related Commands	nv action import system security certificate {cacert id}
Notes	

5.4.8.3 nv show system security crl

	nv show system security crl Display owned certificates.
Syntax Description	N/A

Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$nv show system security crl ID Issuer CRL Number Last Update Count Revoked Summary ---------- ------- ---------- ---------- ----- ------- ---------- crl_01 Root CA 1 Mar 26 09:04:38 2025 1 1 Installed: gnmi-server
REST API	GET https://<ip>/nvue_v1/system/security/crl/
Related Commands	nv action import system security crl {crl id}
Notes	

5.4.8.4 nv action delete system security ca-certificate

	nv action delete system security ca-certificate <cacert-id> Delete system security CA certificate.	
Syntax Description	cacert-id	CA certificate ID removed during import
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action delete system security ca-certificate {cacert id}	
REST API	DELETE https://<ip>/nvue_v1/system/security/ca-certificate/{cacert id}	
Related Commands	nv action import system security ca-certificate {cacert id}	
Notes		

5.4.8.5 nv action delete system security certificate

	nv action delete system security certificate <cert-id> Delete system security CA certificate.	
Syntax Description	cert-id	Certificate ID removed during import
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action delete system security certificate {cert id}	
REST API	DELETE https://<ip>/nvue_v1/system/security/certificate/{cert id}	
Related Commands	nv action import system security certificate {cert id}	
Notes		

5.4.8.6 nv action import system security ca-certificate

	nv action import system security ca-certificate <cacert-id> <uri {remote-url} | data> <remote-url | cacert-data> [external-ca] Import system security CA certificate bundle.	
Syntax Description	cacert-id	Unique CA Certificate ID that was named by the user
	uri	A local/remote URI from where the certificate file (containing the public-key) can be retrieved. Supports: ftp, scp and sftp (e.g., scp://user[:password]@hostname/path/filename, file:///absolute-path/filename)
	data	The raw data bytes (e.g., PEM string) of the CA certificates bundle.
	remote-url	A local/remote URI from where the certificate file (containing the CA certificate bundle) can be retrieved.
	external_ca	Optional parameter to import certificate without appending it to system CA certificates bundle at `/etc/ssl/certs/ca-certificates.crt`.
Default	N/A	
History	25.02.2002	

Example	
	`admin@nvos:~$ nv action import system security ca-certificate tls-cert-1 uri scp://user:pass@1.2.3.4:/ca-cert.crt`
	`admin@nvos:~$ nv action import system security ca-certificate tls-cert-1 uri file:///ca-cert.cr`
	`admin@nvos:~$ nv action import system security ca-certificate tls-cert-1 data "<CA-certificate>"`
	`admin@nvos:~$ nv action import system security ca-certificate tls-cert-1 uri scp://user:pass@1.2.3.4:/ca-cert.crt external`	

REST API	POST https://<ip>/nvue_v1/system/security/ca-certificate/{cacert-id}	
Related Commands	nv action delete system security ca-certificate {cacert-id} nv show sys security ca-certificate	
Notes		

5.4.8.7 nv action import system security certificate

	nv action import system security certificate <cert-id> <uri-public-key <uri-path> uri-private-key <uri-path> | uri-bundle <uri-path> [passphrase] | data> Import system security certificate.	
Syntax Description	cert-id	Unique Certificate ID that was named by the user
	uri-public-key	A local/remote URI from where the public key file can be retrieved.

	uri-private-key	A local/remote URI from where the private key file can be retrieved.
	uri-bundle	A local/remote URI from where the certificate file containing the certificate bundle can be retrieved. Needs to be in .p12 format.
	uri-path	A local/remote URI from where the certificate file (containing the CA certificate bundle) can be retrieved.
	data	The raw data bytes (e.g., PEM string) of the certificates bundle
	passphrase	Optional passphrase if certificate bundle is passphrase protected
Default	N/A	
History	25.02.2002	

Example	`admin@nvos:~$ nv action import system security certificate tls-cert-1 passphrase mypassphrase uri-bundle scp://user:pass@1.2.3.4:/opt/certs/cert.p12`
	`admin@nvos:~$ nv action import system security certificate tls-cert-1 passphrase mypassphrase uri-bundle file:///opt/certs/cert.p12`
	`admin@nvos:~$ nv action import system security certificate tls-cert-1 uri-public-key scp://user:pass@1.2.3.4:/opt/certs/public uri-private-key scp://user:pass@1.2.3.4:/opt/certs/private`
	`admin@nvos:~$ nv action import system security certificate tls-cert-1 uri-public-key file:///opt/certs/public uri-private-key file:///opt/certs/private`
	`admin@nvos:~$ nv action import system security certificate tls-cert-1 data "<certificate>"`	

REST API	POST https://<ip>/nvue_v1/system/security/certificate/{cert-id}	
Related Commands	nv action delete system security certificate nv show sys security certificate	
Notes		

5.4.8.8 nv action import system security crl

	nv action import system security crl <crl-id> <uri {remote-url}> Import system security CRL.	
Syntax Description	crl-id	Unique CRL ID that was named by the user
	uri	A local/remote URI from where the certificate file (containing the public-key) can be retrieved. Supports: ftp, scp and sftp (e.g., scp://user[:password]@hostname/path/filename, file:///absolute-path/filename)

	remote-url	A local/remote URI from where the certificate file (containing the CA certificate bundle) can be retrieved.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action import system security crl crl-1 uri scp://user:pass@1.2.3.4:/crl_file.crl admin@nvos:~$ nv action import system security crl crl-1 uri file:///crl.crt	
REST API	POST https://<ip>/nvue_v1/system/security/crl/{crl-id}	
Related Commands	nv action delete system security crl {crl-id} nv show system security crl	
Notes		

5.4.8.9 nv action delete system security crl

	nv action delete system security crl <crl-id> Delete system security CRL certificate.	
Syntax Description	crl-id	CRL ID removed during import
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action delete system security crl crl_01	
REST API	DELETE https://<ip>/nvue_v1/system/security/crl/{crl-id}	
Related Commands	nv action import system security crl {crl id}	
Notes		

5.5 Control and Power

NVOS allows to perform actions such as rebooting the system, initiating a power-cycle, and retrieving information about past reboots.

5.5.1 Possible Reboot Causes

- "Power Loss"
- "Thermal Overload: CPU"
- "Thermal Overload: ASIC"
- "Thermal Overload: Other"
- "Insufficient Fan Speed"

- "Watchdog"
- "Hardware - Other"
- "BIOS"
- "CPU"
- "Long push button"
- "Short push button"
- "Reset from ASIC"

Operations like firmware installation (nv action install platform firmware files) may change the behavior of the next system reboot to do a full power-cycle to apply the newly burned firmware.

5.5.2 Control and Power Commands

- Control and Power Commands

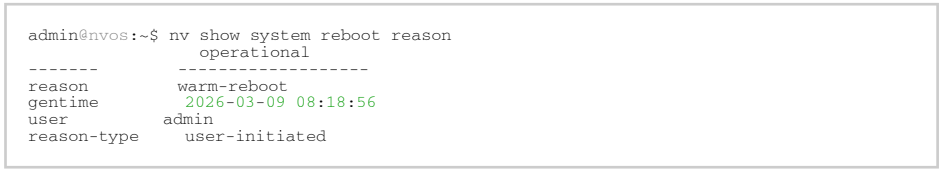
5.5.3 Control and Power Commands

5.5.3.1 nv show system reboot

	nv show system reboot Show system reboot information.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show system reboot operational applied --------- ---------- ------- [history] reason gentime 2022_06_02_07_32_05 reason Hardware - Other (Reset from ComEx) user N/A
REST API	GET https://<ip>/nvue_v1/system/reboot
Related commands	

Notes	

5.5.3.2 nv show system reboot reason

	nv show system reboot reason Show the system reboot reason.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.8008 Added `reason-type` field
Example	
REST API	GET https://<ip>/nvue_v1/system/reboot/reason
Related commands	
Notes	

5.5.3.3 nv show system reboot history

	nv show system reboot history Show the system reboot history.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.8008 Added `reason-type` field
Example	
REST API	GET https://<ip>/nvue_v1/system/reboot/history
Related commands	
Notes	

5.5.3.4 nv show system reboot counters

	nv show system reboot counters Show the number of reboots of different types (grouped by the openconfig defined types from `COMPONENT_REBOOT_REASON`).

Syntax Description	N/A
Default	N/A
History	25.02.8008
Example	admin@nvos:~$ nv show system reboot counters reason-type count ---------- -------- user-initiated 7 power-failure 3 critical-error 5 total 15
REST API	GET https://<ip>/nvue_v1/system/reboot/counters
Related commands	nv show system reboot history
Notes	

5.5.3.5 nv action reboot system

	nv action reboot system [mode {halt | cold | immediate}] [force] Reboot switch system.	
Syntax Description	flag	• halt: Shuts down the system • cold: Allow a controlled shutdown by notifying all running processes that the system is going down • immediate: Reboot system immediately without notifying any running processes
	force	Force the action without asking for user confirmation.
Default	N/A	
History	25.02.2002 25.02.5002 Updated reboot modes	
Example	admin@nvos:~$ nv action reboot system Configuration has been modified, but not saved. Type [y] to reboot the system without saving configuration. Type [N] to abort. Do you want to continue? [Y/N] N System reboot aborted by user admin@nvos:~$ nv action reboot system mode halt Type [y] to halt the system. Type [N] to abort. WARNING: This operation will shut down the system. You will NOT be able to turn on the system remotely. Do you want to continue? [y/N]	
REST API	POST https://<ip>/nvue_v1/system	
Related commands	nv action install platform firmware files	

Notes	• The prompt will be displayed only if there is unsaved configuration on the switch. Otherwise, the reboot will be performed as usual. • Using the halt flag will show the prompt unless force flag is passed. • Using the immediate flag will not trigger the firmware upgrade flow that part of normal reboot. • If the switch is in a fatal state, it will exit that state once it comes up. • `nv action reboot system` will still work as usual which is equivalent to `nv action reboot system mode cold`

5.5.3.6 nv show platform ps-redundancy

	nv show platform ps-redundancy Shows power supply redundancy policy
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show platform ps-redundancy operational applied ------- ------------- ------------- policy grid-redundant grid-redundant min-required 8
REST API	GET https://<ip>/nvue_v1/platform/ps-redundancy
Related Commands	nv set platform ps-redundancy policy nv show system health
Notes	Supported only in systems that do not require all the power supplies to be connected.

5.5.3.7 nv set platform ps-redundancy policy

	nv set platform ps-redundancy policy {no-redundancy | ps-redundant | grid-redundant} Sets system power redundancy policy. When the power redundancy policy requirements are not met a health issue is raised	
Syntax Description	no-redundancy	No power supply is reserved. The redundancy is not enabled.
	ps-redundant	One power supply unit is redundant to the rest. The system can work with one less power supply unit.
	grid-redundant	The power supplies are split into two logical power supply grids, each grid contains the minimum power supplies required by the system. The power supplies on the left side of the switch are one grid, and the power supplies on the right switch are the other. The system can work if all the power supplies of one grid fail.
Default	grid-redundant	
History	25.02.2002	
Example	nv set platform ps-redundancy policy grid-redundant	
REST API	PATCH https://<ip>/nvue_v1/ platform/ps-redundancy	

Related Commands	nv show system health nv show platform ps-redundancy
Notes	Supported only in systems that do not require all the power supplies to be connected.

5.6 DNS Server

Domain Name System (DNS) is an essential component of modern networking, translating human-readable domain names into IP addresses that systems can understand. Proper DNS configuration ensures seamless connectivity and efficient resolution of network requests.

NVOS provides an ability to manage DNS servers, allowing users to add, configure, and view DNS server configurations as needed.

5.6.1 Supported Configurations and Limitations

- Both IPv4 and IPv6 unicast DNS server addresses configurations are supported
- NVOS obtains dynamic DNS entries from DHCP server by default (this will be preserved if no static DNS entries are configured)
- Dynamic DNS configuration will not work if management interface uses static IP configuration
- Linux allows for a maximum of three DNS servers to be configured at the same time

5.6.2 DNS Server Commands

5.6.3 DNS Server Commands

5.6.3.1 nv show system dns

	nv show system dns Display DNS configuration.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show system dns operational applied -------- ----------- ----------- [server] 8.8.8.8 8.8.8.8 [server] 10.7.77.135 10.7.77.135 [server] 10.7.77.192 10.7.77.192
REST API	GET https://<ip>/nvue_v1/system/dns
Related Commands	nv show system dns server nv set/unset system dns server
Notes	

5.6.3.2 nv show system dns server

	nv show system dns server Display list of configured DNS servers.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show system dns server DNS Server ----------- 8.8.8.8 10.7.77.135 10.7.77.192	
REST API	GET https://<ip>/nvue_v1/system/dns/server	
Related Commands	nv set/unset system dns server	
Notes		

5.6.3.3 nv set/unset system dns server

	nv set system dns server {dns-server-ip} nv unset system dns server {dns-server-ip} Update the DNS server configuration.	
Syntax Description	dns-server-ip	IPv4 or IPv6 unicast address of a DNS server
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system dns server 249.1.1.1 Error: '249.0.0.0' is not a 'dns-server-ip-address'. IP address must be unicast IPv4/ IPv6 address. admin@nvos:~$ nv set system dns server 8.8.8.8 admin@nvos:~$ nv set system dns server 1.1.1.1 admin@nvos:~$ nv set system dns server 10.7.77.192 admin@nvos:~$ nv set system dns server 10.7.77.135 admin@nvos:~$ nv config apply The maximum number 3 of DNS servers exceeded. admin@nvos:~$ nv unset system dns server 10.7.77.135 admin@nvos:~$ nv unset system dns server admin@nvos:~$ nv unset system dns	

REST API	PATCH https://<ip>/nvue_v1/system/dns/server/{dns-server-ip}
Related Commands	nv show system dns server
Notes	The maximum number of DNS servers is limited to three by Linux.

5.7 Documentation

Users can view and upload any of the documentation files provided by the NVOS as part of its release, including the EULA, User Manual, Release Notes, and Open Source Licenses.

5.7.1 Documentation Commands

5.7.2 Documentation Commands

5.7.2.1 nv show system documentation

	nv show system documentation Display system document list.	
Syntax Description	files	Displays system document list in brief mode.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show system documentation Name Type Path ---------- ---------- NVOS_EULA.pdf EULA /usr/share/nginx/html/system_documents/eula/NVOS_EULA.pdf NVOS_NVL_Release_Notes.pdf Release notes /usr/share/nginx/html/system_documents/release_notes/NVOS_NVL_Release_Notes.pdf NVOS_NVL_User_Manual.pdf User manual /usr/share/nginx/html/system_documents/user_manual/NVOS_NVL_User_Manual.pdf Open_Source_Licenses.txt Open source licenses /usr/share/nginx/html/system_documents/open_source_licenses/Open_Source_Licenses.txt	
Related Commands	nv action upload system documentation files	
Notes		

5.7.2.2 nv action upload system documentation files

	nv action upload system documentation files <file-name> <remote-url> Upload system document to remote server

Syntax Description	file-name	Document to be uploaded.
	remote-url	Destination image file name Remote url path to upload a file to. Format: [protocol]://username[:password]@hostname/path/filename Supported protocols: SCP, FTP, SFTP, and HTTPS
Default	N/A	
History	25.02.2002 25.02.4002: Added HTTPS support in remote-url	
Example	admin@nvos:~$ nv action upload system documentation files NVOS_EULA.pdf scp://user:password@10.1.12.20/tmp/NVOS_EULA.pdf	
Related Commands	nv show system documentation	
Notes		

5.8 Hostname

NVOS provides the ability to set a hostname to the switch. Make sure that the hostname is unique and descriptive.
There are two option to set the hostname: Dynamically through DHCP or Statically in NVUE.

5.8.1 Hostname from DHCP

By default, DHCP is enabled, device receives the DHCP Hostname option inside the response and set the device hostname.

By default, DHCP is enabled for both interfaces that may receive hostname update during run. NVOS is always updating to the latest hostname received by DHCP.
It is possible to disable hostname for a certain interface by executing the following:

```
admin@nvos:~$ nv set interface eth0 ip dhcp-client set-hostname disabled
admin@nvos:~$ nv config apply
```

Do not use an underscore (_), apostrophe ('), or non-ASCII characters in the hostname.

The hostname convention need to follow "idn-hostname" as defined by either RFC 1123 as for hostname, or an internationalized hostname as defined by RFC 5890, section 2.3.2.3 [RFC5890]

5.8.2 Static Hostname

To change the hostname, run the following:

```
admin@nvos:~$ nv set system hostname leaf01
admin@nvos:~$ nv config apply
```

The command prompt in the terminal does not reflect the new hostname until either logging out of the switch or starting a new shell.

5.8.3 Hostname Commands

- Hostname Commands

5.8.4 Hostname Commands

5.8.4.1 nv set/unset system hostname

	nv set/unset system hostname {hostname} Set/unset the hostname of the switch.	
Syntax Description	hostname	The new hostname to set
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system hostname switch01 admin@nvos:~$ nv unset system hostname	
REST API	PATCH https://<ip>/nvue_v1/system	
Related Commands	nv show system	
Notes		

5.9 Link Layer Discovery Protocol

Link Layer Discovery Protocol (LLDP) is a standard protocol that lets network devices share basic information (such as name, port, and capabilities) with directly connected neighbors. It helps with network discovery and troubleshooting without affecting traffic.

In NVOS, LLDP is supported on management ports only.

5.9.1 LLDP State

To check LLDP state, run the following command:

```
admin@nvos:~$ nv show system lldp

                      operational   applied
--------------------  -----------   -------
state                 enabled       enabled
tx-interval           30            30
tx-hold-multiplier    4             4
```

5.9.2 LLDP Neighbor Information

To check LLDP neighbor informaion, run the following command:

```
admin@nvos:~$ nv show interface eth0 lldp neighbor
Neighbor                    Remote IP    Model    SW Version        Remote Port
--------------------------  -----------  -------  ----------------  -----------
MTL-S-F2-LAB-ADVG-SW-4-12   10.60.4.12                              11
```

5.9.3 LLDP Commands

- Link Layer Discovery Protocol Commands

5.9.4 Link Layer Discovery Protocol Commands

5.9.4.1 nv set/unset system lldp

	nv set system lldp [state {enabled,disabled} | tx-interval {number} | tx-hold-multiplier {number}] nv unset system lldp [state {enabled,disabled} | tx-interval {number} | tx-hold-multiplier {number}] Update LLDP global configurations.	
Syntax Description	state	LLDP state configuration.
	tx-interval	Change transmit delay.
	tx-hold-multiplier	TTL of transmitted packets is calculated by multiplying the tx-interval by the given factor.
Default	state	enabled
	tx-interval	30
	tx-hold-multiplier	4
History	25.02.2002	
Example	admin@nvos:~$ nv set system lldp state disabled admin@nvos:~# nv set system lldp state tx-interval 50	
REST API	PATCH https://<ip>/nvue_v1/system/lldp	
Related Commands	nv show system lldp	
Notes		

5.9.4.2 nv show system lldp

	nv show system lldp Display LLDP global configurations.	
Syntax Description	N/A	
Default	N/A	

History	25.02.2002
Example	admin@nvos:~$ nv show system lldp operational applied ------------------ ----------- ------- state enabled enabled tx-interval 30 30 tx-hold-multiplier 4 4
REST API	GET https://<ip>/nvue_v1/system/lldp
Related Commands	nv set system lldp
Notes	

5.9.4.3 nv show interface lldp

	nv show interface <interface-id> lldp Show preview details of interface neighbor.	
Syntax Description	interface-id	interface name
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show interface eth0 lldp operational applied ---------- ------------------------- ------- [neighbor] MTL-S-F2-LAB-ADVG-SW-4-12	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/lldp	
Related Commands	nv show interface {interface-id} lldp neighbor	
Notes	This command is just for mgmt interfaces.	

5.9.4.4 nv show interface lldp neighbor

	nv show interface <interface-id> lldp neighbor Show details of interface neighbor.	
Syntax Description	interface-id	interface name
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show interface eth0 lldp neighbor Neighbor Remote IP Model SW Version Remote Port ------------------------ ---------- ------- ---------- ---------- MTL-S-F2-LAB-ADVG-SW-4-12 10.60.4.12 11	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/lldp/neighbor	
Related Commands	nv show interface <interface-id> lldp neighbor <neighbor-id>	
Notes	This command is just for mgmt interfaces.	

5.9.4.5 nv show interface lldp neighbor id

	nv show interface <interface-id> lldp neighbor <neighbor-id> Show extend details of the neighbor	
Syntax Description	interface-id	Interface name
	neighbor-id	Neighbor name
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show interface eth0 lldp neighbor MTL-S-F2-LAB-ADVG-SW-4-12
	                          operational
	------------------------ --
	age 3581
	bridge
	  untagged 59
	chassis
	  chassis-id 1c:98:ec:45:3e:c0
	  management-address-ipv4 10.60.4.12
	  management-address-ipv6 fdfd:fdfd:7:145::1000:4998
	  system-name MTL-S-F2-LAB-ADVG-SW-4-12
	  system-description HP J9775A 2530-48G Switch, revision YA.16.11.0015, ROM
	YA.15.20 ...
	port
	  ttl 120
	  name 11
	  type local
	  description 11
	  pmd-autoneg
	    [advertised] 0
	    [advertised] 1
	    [advertised] 2
	    mau-oper-type 1000BaseTFD - Four-pair Category 5 UTP, full duplex mode
	
	lldp-med
	  device-type</pre>	

REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/lldp/neighbor/{neighbor-id}	
Related Commands	nv show interface {interface-id} lldp neighbor	
Notes	This command is just for mgmt interfaces.	

5.9.4.6 nv show interface lldp

	nv show interface lldp Show neighbor details of all interfaces.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show interface lldp
	Interface Speed Type Neighbor Remote Port
	--------- ------ -------- ------------------------ -----------
	eth0 1G eth MTL-S-F2-LAB-ADVG-SW-4-12 11
	ib0 ipoib
	lo loopback
	sw1p1 200G ib
	sw1p2 200G ib
	sw2p1 ib
	... ...</pre>	

REST API	GET https://<ip>/nvue_v1/interface/lldp	
Related Commands	nv show interface {interface-id} lldp	
Notes		

5.10 Management Interfaces

Management interfaces are used in order to provide access to switch management user interfaces (e.g., CLI, openAPI).

NVIDIA switches support out-of-band (OOB) dedicated interfaces (e.g., eth0). In addition, most NVIDIA switches feature a serial port that provides access to the CLI only.

5.10.1 Configuring Management Interfaces with Static IP Addresses

The management interface uses DHCP for addressing by default.

To set a static IP address (for example):

```
admin@nvos:~$ nv set interface eth0 ip address 192.0.2.42/24
admin@nvos:~$ nv set interface eth0 ip gateway 192.0.2.1
admin@nvos:~$ nv config apply
```


To help setups with static IP to detect new devices, configuring static IP address will trigger unsolicited announcement messages to the gateway in order to reveal the devices's MAC address. This also applies to static IPv6 addresses.
Physically disconnecting and connecting the cables will also trigger the same messages when static IP addresses are configured.

5.10.2 Configuring IPv6 Address on the Management Interface

1. Set a static IPv6 address. Run:

   ```
   admin@nvos:~$ nv set interface eth0 ip address fdfd:fdfd:7:145:9a03:9bff:fe6b:6ac/64
   admin@nvos:~$ nv config apply
   ```

2. Verify the IPv6 address is configured correctly. Run:

   ```
   admin@nvos:~$ nv show interface eth0
   ```

5.10.3 Default Gateway

To configure manually the default gateway, use the “nv set interface eth0 ip gateway” command, with “0.0.0.0” as prefix and mask. The next-hop address must be within the range of one of the IP interfaces on the system.

5.10.4 IP DHCP Client

To enable DHCP client and DHCPv6 client state on Management (eth0, eth1) interfaces use the following commands:

1. nv set interface <interface-id> ip dhcp-client state <enabled | disabled>
2. nv set interface <interface-id> ip dhcp-client6 state <enabled | disabled>

> ⚠ Enabling DHCP client on management interface (eth0, eth1) automatically enables DHCPv6 client and vice-versa.

5.10.5 Management Interface Commands

5.10.5.1 nv show interface

	nv show interface [interface-id] Displays details of an IPoIB/eth0 interface. If not interface is selected, summary of attributes of all interfaces will be displayed	
Syntax Description	interface-id	Name of the interface to display (e.g., eth0)
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output 25.02.70xx Updated command output	

Example

```
admin@nvos:~$ nv show interface ib0
                        operational
applied
----------------------  ------------------------------------------------------------
--------
ipv4
  dhcp-client
    state          disabled                                                  disabled
    set-hostname   enabled                                                   enabled
  [gateway]
  arp-timeout      1800                                                      1800
ipv6
  dhcp-client
    state          disabled                                                  disabled
    set-hostname   enabled                                                   enabled
  [gateway]
  autoconf         disabled                                                  disabled

link
  auto-negotiate
enabled
  duplex
full
  speed
auto
  mac                    80:00:00:02:fe:80:00:00:00:00:00:00:90:0a:84:03:00:76:a8:00
  mtu                    2044
2044
  state                  up                                                        up
ifindex                  8
counters
  in-bytes         0 Bytes
  in-pkts          0
  in-drops         0
  in-errors        0
  out-bytes        0 Bytes
  out-pkts         0
  out-drops        0
  out-errors       0

type                     ipoib
ipoib
```

```
admin@nvos:~$ nv show interface
Interface  State  Speed  MTU    Type      Description  IB speed  IB subnet
Logical state  Physical state  Summary
---------  -----  -----  -----  --------  -----------  --------  ------------------
-------------  --------------  --------------------------------------------------
eth0       up     1G     1500   eth
IPv4 Address:                          10.7.89.7/20

IPv6 Address: fdfd:fdfd:7:80:eaeb:d3ff:fe4b:70b8/64
ib0        up             2044   ipoib
lo         up             65536  loopback
IPv4 Address:                          127.0.0.1/16

IPv6 Address:                              ::1/128
sw1p1      down                   ib                                  infiniband-default  Down
Polling
sw1p2      down                   ib                                  infiniband-default  Down
Polling
sw2p1      down                   ib                                  infiniband-default  Down
Polling
. . .
sw32p1     up     400G   4096   ib                       ndr
infiniband-default  Active         LinkUp
sw32p2     up     400G   4096   ib                       ndr
infiniband-default  Active         LinkUp
```

```
admin@nvos:~$ nv show interface eth0

                         operational                         applied
-----------------------  ----------------------------------
---------------------------------
[acl]                    ACL_MGMT_INBOUND_CP_DEFAULT         ACL_MGMT_INBOUND_CP_DEFAULT
[acl]                    ACL_MGMT_INBOUND_CP_DEFAULT_IPV6
ACL_MGMT_INBOUND_CP_DEFAULT_IPV6
[acl]                    ACL_MGMT_INBOUND_DEFAULT            ACL_MGMT_INBOUND_DEFAULT
[acl]                    ACL_MGMT_INBOUND_DEFAULT_IPV6       ACL_MGMT_INBOUND_DEFAULT_IPV6
[acl]                    ACL_MGMT_OUTBOUND_CP_DEFAULT        ACL_MGMT_OUTBOUND_CP_DEFAULT
[acl]                    ACL_MGMT_OUTBOUND_CP_DEFAULT_IPV6
ACL_MGMT_OUTBOUND_CP_DEFAULT_IPV6
lldp
  [neighbor]

vrf                          default                             default
ipv4
  dhcp-client
    state                    enabled                             enabled
    set-hostname             enabled                             enabled
    lease
      fixed-address          10.7.148.195
      subnet-mask            255.255.248.0
      routers                10.7.144.1
      broadcast-address      10.7.151.255
```

```
      filename                /uefiboot/bootx64-signed.efi
      server-name             10.7.77.122
      dhcp-lease-time         21600
      dhcp-message-type       5
      host-name               nvos
      domain-name             mtl.labs.mlnx
      renew                   2025-09-09 05:17:56
      rebind                  2025-09-09 07:49:37
      expire                  2025-09-09 08:34:37
      [domain-search]         lab.nvidia.com.
      [domain-search]         mlnx.
      [domain-search]         mtl.labs.mlnx.
      [domain-search]         mtr.labs.mlnx.
      [domain-search]         nbulabs.nvidia.com.
      [domain-search]         nvidia.com.
      [domain-name-servers]   10.7.77.135
      [domain-name-servers]   10.7.77.192
      [domain-name-servers]   10.198.0.169
      [domain-name-servers]   10.211.0.124
      [ntp-servers]           10.7.77.134
      [ntp-servers]           10.7.77.135
  [address]                   10.7.148.195/21
  [gateway]                   10.7.144.1
  arp-timeout                 1800                                    1800
ipv6
  dhcp-client
    state                     enabled                                 enabled
    set-hostname              enabled                                 enabled
    lease
      client-id               0:3:0:1:e0:9d:73:c:5d:8c
      server-id               0:1:0:1:25:f2:90:34:0:50:56:8b:e4:f9
      [ia-na]                 73:0c:5d:8c
  [address]                   fdfd:fdfd:7:145::1000:4751/128
  [address]                   fdfd:fdfd:7:145:e29d:73ff:fe0c:5d8c/64
  [gateway]
  autoconf                    enabled                                 enabled
link
  auto-negotiate          enabled                             enabled
  duplex                                                      full
  speed                                                       1G
  mac                     c4:70:bd:8f:51:32
  mtu                     1500                                1500
  state                   down                                up
ifindex                   2
counters
  in-bytes                    14.96 MB
  in-pkts                     113008
  in-drops                    0
  in-errors                   0
  out-bytes                   4.06 MB
  out-pkts                    44178
  out-drops                   0
  out-errors                  0

type                      eth                                 eth
```

REST API	GET https://<ip>/nvue_v1/interface
Related Commands	nv set interface nv show interface
Notes	

5.10.5.2 nv show interface link

	nv show interface <interface-id> link {brief | state} Displays link information of an IPoIB/eth0 interface.	
Syntax Description	interface-id	Name of the interface to display (e.g., eth0)
	brief	
	state	Show only the data relating to state
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output 25.02.70xx Removed counters	

Example	admin@nvos:~$ nv show interface eth0 link operational applied ---------- ---------- ------- auto-negotiate enabled enabled duplex full speed 1G mac c4:70:bd:8f:51:32 mtu 1500 1500 state down up admin@nvos:~$ nv show interface eth0 link state operational applied -- ---------- ------- up
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/link
Related Commands	nv show interface nv set interface link nv unset interface link
Notes	

5.10.5.3 nv show interface link phy health

	nv show interface <interface-id> link phy health Displays link PHY health details.	
Syntax Description	interface-id	Name of the interface to display (e.g., eth0)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface sw68p2 link phy health operational ------------------------ ----------------- time-since-last-clear-min 396.61 symbol-errors 0 symbol-ber 15E-255 raw-ber 3E-8 phy-received-bits 20227291902548640 effective-errors 0 effective-ber 15E-255 Lane Stats ============= Lane raw-ber phy-raw-errors ---- ------- -------------- 0 3E-8 337813882 1 0E-0 0 2 0E-0 0 3 0E-0 0 4 0E-0 0 5 0E-0 0 6 0E-0 0 7 0E-0 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/link/phy/health	
Related Commands	nv show interface <interface-id> link phy health histogram	
Notes		

5.10.5.4 nv show interface link phy health histogram

	nv show interface <interface-id> link phy health histogram Displays link PHY health histogram (rs-fec corrected errors bin) details.	
Syntax Description	interface-id	Name of the interface to display (e.g., eth0)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface sw68p2 link phy health histogram rs-fec-corrected-errors ========== Bin count --- ----- 0 3708836002770 1 288537427 2 24833730 3 3374 4 139 5 0 6 0 7 0 8 0 9 0 10 0 11 0 12 0 13 0 14 0 15 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/link/phy/health/histogram	
Related Commands	nv show interface <interface-id> link phy health	
Notes		

5.10.5.5 nv show interface link phy detail

	nv show interface <interface-id> link phy detail Displays link phy detail details	
Syntax Description	interface-id	Name of the interface to display (e.g., eth0)
Default	N/A	
History	25.02.70xx	

Example

```
admin@nvos:~$ nv show interface sw68p2 link phy detail

operational

-------------------------------------
-------------------------------------------------------------------------------------------------
--------------------------
loopback-mode                           NONE/NONE/NONE/
NONE
fec-mode-request                        64/64/64/
64

cable-proto-cap-ext                     2:11,11/2:11,11/2:11,11/
2:11,11

pd-fsm-state                            0/0/0/0

eth-an-fsm-state                        0/0/0/
0

phy-hst-fsm-state                       LINKUP/LINKUP/LINKUP/
LINKUP
psi-fsm-state                           IDLE/IDLE/IDLE/
IDLE

retran-mode-request                     0/0/0/
0

retran-mode-active                      0/0/0/
0

profile-fec-in-use                      5/5/5/
5

pd-link-width-enabled                   0/0/0/
0

pd-link-speed-enabled                   1:3690988944/1:1476396432/1:1677723072/
1:1677723040
phy-hst-link-width-enabled              0/0/0/
0

phy-hst-link-speed-enabled              1:3690987664/1:1476395152/1:1677723040/
1:1677723024
phy-manager-state                       ACTIVE/ACTIVE/ACTIVE/
ACTIVE

phy-manager-link-width-enabled          1/1/1/
1

phy-manager-link-proto-enabled          4:800000,400000,200000,10000/
4:800000,400000,200000,10000/4:800000,400000,200000,10000/4:800000,400000,200000,10000
core-to-phy-link-width-enabled          1/1/1/
1

core-to-phy-link-proto-enabled          1:800000/1:800000/1:800000/
1:800000

sync-header-error-counter               0

port-local-physical-errors
0

port-malformed-packet-errors
0

plr-rcv-codes
1857709441290

plr-rcv-codes-err
0

plr-rcv-uncorrectable-code              0

plr-xmit-codes
1857709696712

plr-xmit-retry-codes
0

plr-xmit-retry-events                   0

plr-sync-events
0

plr-codes-loss
                0

plr-xmit-retry-events-within-t-sec-max
0

plr-bw-loss-percent
0.0

rq-general-error                        0

ib-phy-fsm-state                        DISABLED/DISABLED/DISABLED/
DISABLED
zero-hist
5
```

```
successful-recovery-events
0

port-buffer-overrun-errors
0

port-dlid-mapping-errors
      292

port-vl-mapping-errors
0

port-looping-errors
0

port-inactive-discards                  0

port-neighbor-mtu-discards
0

unintentional-link-down-events
0

intentional-link-down-events
0

time-in-last-logic-recovery-event
0

time-in-last-serdes-eq-recovery-event   0


time-since-last-recovery
0

last-logic-recovery-attempts
0

last-serdes-eq-recovery-attempts        0

time-between-last-two-recoveries
0

total-successful-recovery-events
0

linkdown-reason-code-local              0/0/0/
0

linkdown-reason-status-local            NO_LINK_DOWN_INDICATION/NO_LINK_DOWN_INDICATION/
NO_LINK_DOWN_INDICATION/NO_LINK_DOWN_INDICATION
linkdown-reason-code-remote             37/37/37/
37

linkdown-reason-status-remote           PEER_SIDE_DOWN_DUE_TO_RESET_EVENT/
PEER_SIDE_DOWN_DUE_TO_RESET_EVENT/PEER_SIDE_DOWN_DUE_TO_RESET_EVENT/
PEER_SIDE_DOWN_DUE_TO_RESET_EVENT
```

REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/link/phy/detail
Related Commands	
Notes	

5.10.5.6 nv set/unset interface link connection-mode

	nv set interface <interface-id> link connection-mode {value} nv unset interface <interface-id> link connection-mode Change the interface connection-mode value according to his peer connectivity.	
Syntax Description	interface-id	Name of the interface to display.
	value	The connection-mode values: xdr/ndr
Default	XDR	
History	25.02.2002	
Example		
REST API	PATCH https://<ip>/nvue_v1/interface/{interface-id}/link	

```
$ nv set interface swA1p1 link connection-mode ndr
```

Related Commands	nv show interface swA1p1 link
Notes	This command is only supported on Q3200-RA system.

5.10.5.7 nv show interface counters

	nv show interface <interface-id> counters Displays counters information of an IPoIB/eth0 interface.	
Syntax Description	interface-id	Name of the interface to display (e.g., eth0, swB14p2, acp142)
Default	N/A	
History	25.02.70xx	
Example		

```
admin@nvos:~$ nv show interface swB14p2 counters


                                operational
------------------------------  -----------
in-bytes                        2051136 Bytes
in-pkts                         7122
in-drops                        0
in-errors                       0
out-bytes                       2051136 Bytes
out-pkts                        7122
out-drops                       0
out-errors                      0
out-unicast-pkts                7122
out-multicast-pkts              0
in-unicast-pkts                 7122
in-multicast-pkts               0
buffer-overrun-errors           0
out-wait                        0



Error Counters
==================
    Counter           Receive  Transmit
    ----------------  -------  --------
    icrc-errors        0        n/a
    symbol-errors      0        n/a
    tx-parity-errors   n/a      0


Drop Counters
================
    Counter    Receive  Transmit
    ---------  -------  --------
    qp1-drops  0        0


Fast Recovery Counters
=========================
    Trigger          Consecutive normal windows  Warning events  Error events
    ---------------  --------------------------  --------------  ------------
    credit-watchdog  0                           0               0
    effective-ber    0                           0               0
    raw-ber          0                           0               0
    symbol-ber       0                           0               0
```

REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters
Related Commands	nv show interface
Notes	

5.10.5.8 nv show interface counters link

	nv show interface <interface-id> counters link Displays link-level counters for an interface, including physical and integrity errors.

Syntax Description	interface-id	Name of the interface to display (e.g., eth0, swB14p2, acp142)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface swB14p2 counters link operational ------------------------------ ----------- carrier-down-count 0 error-recovery 0 port-rcv-remote-physical-errors 0 port-rcv-switch-relay-errors 0 port-rcv-constraint-errors 0 local-integrity-errors 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters/link	
Related Commands	nv show interface counters	
Notes		

5.10.5.9 nv show interface <id> lldp neighbor <neighbor-id>

	nv show interface <interface-id> neighbor <neighbor-id> Displays the interface lldp neighbor details	
Syntax Description	interface-id neighbor-id	Name of the interface to display (e.g., eth0) LLDP neighbor of an interface
Default	N/A	
History	25.02.70xx	
Example	nv show interface eth0 lldp neighbor MTL-S-F-3-LAB-SW144 operational -------------------- ---------- age 15286 bridge untagged 59 chassis chassis-id 44:5b:ed:4e:2b:00 system-name MTL-S-F-3-LAB-SW144 system-description HP J9775A 2530-48G Switch, revision YA.16.11.0021, ROM YA.15.20 (/ws/swbuildm/rel_beluru_qaoff/code/build/lakes(swbuildm_rel_beluru_qaoff_rel_beluru)) [capability] is-bridge port ttl 120 name 31 type local description 31 pmd-autoneg [autoneg] is-supported [autoneg] is-enabled [advertised] 0 [advertised] 1 [advertised] 2 mau-oper-type 1000BaseTFD - Four-pair Category 5 UTP, full duplex mode lldp-med device-type Management Interface Info ========== Index Remote IP Remote IPV6 ----- ---------- ---------- 1 10.60.6.144	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/lldp/neighbor/{neighbor-id}	
Related Commands	nv show interface <interface-id> lldp neighbor	
Notes		

5.10.5.10 nv show interface counters ib

	nv show interface <interface-id> counters ib Displays InfiniBand (IB) interface counters including errors, drops, and fast recovery statistics.	
Syntax Description	interface-id	Name of the interface to display (e.g., swB14p2)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface swB14p2 counters ib Drop Counters Counter Receive Transmit qp1-drops 0 0 Error Counters Counter Receive Transmit icrc-errors 0 n/a symbol-errors 0 n/a tx-parity-errors n/a 0 Fast Recovery Counters Trigger Consecutive normal windows Warning events Error events credit-watchdog 0 0 0 effective-ber 0 0 0 raw-ber 0 0 0 symbol-ber 0 0 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters/ib	
Related Commands	nv show interface counters	
Notes		

5.10.5.11 nv show interface counters ib errors

	nv show interface <interface-id> counters ib errors Displays InfiniBand interface error counters such as ICRC, symbol, and parity errors.	
Syntax Description	interface-id	Name of the interface to display (e.g., swB14p2)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface swB14p2 counters ib errors Counter Receive Transmit icrc-errors 0 n/a symbol-errors 0 n/a tx-parity-errors n/a 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters/ib/errors	
Related Commands	nv show interface counters ib	
Notes		

5.10.5.12 nv show interface counters ib drops

	nv show interface <interface-id> counters ib drops Displays InfiniBand interface drop counters (e.g., QP1 drops for receive and transmit).	
Syntax Description	interface-id	Name of the interface to display (e.g., swB14p2)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface swB14p2 counters ib drops Counter Receive Transmit --------- ------- -------- qp1-drops 0 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters/ib/drops	
Related Commands	nv show interface counters ib	
Notes		

5.10.5.13 nv show interface counters ib fast-recovery

	nv show interface <interface-id> counters ib fast-recovery Displays InfiniBand fast-recovery counters showing link recovery triggers and error statistics.	
Syntax Description	interface-id	Name of the interface to display (e.g., swB14p2)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface swB14p2 counters ib fast-recovery Trigger Consecutive normal windows Warning events Error events --------------- -------------------------- -------------- ------------ credit-watchdog 0 0 0 effective-ber 0 0 0 raw-ber 0 0 0 symbol-ber 0 0 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters/ib/fast-recovery	
Related Commands	nv show interface counters ib	
Notes		

5.10.5.14 nv show interface counters nvl

	nv show interface <interface-id> counters nvl Displays NVLink interface counters including errors, drops, and fast recovery events.	
Syntax Description	interface-id	NVLink interface name (e.g., acp142)
Default	N/A	
History	25.02.70xx	

Example	admin@nvos:~$ nv show interface acp142 counters nvl Drop Counters ================ Counter Receive Transmit --------- ------- -------- qp1-drops 0 0 Error Counters ================== Counter Receive Transmit ---------- ------- -------- icrc-errors 0 n/a symbol-errors 0 n/a tx-parity-errors n/a 0
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters/nvl
Related Commands	nv show interface counters
Notes	

5.10.5.15 nv show interface counters nvl errors

	nv show interface <interface-id> counters nvl errors Displays NVLink interface error counters such as ICRC, symbol, and parity errors.	
Syntax Description	interface-id	NVLink interface name (e.g., acp142)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface acp142counters nvl errors Counter Receive Transmit ---------- ------- -------- icrc-errors 0 n/a symbol-errors 0 n/a tx-parity-errors n/a 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters/nvl/errors	
Related Commands	nv show interface counters nvl	
Notes		

5.10.5.16 nv show interface counters nvl drops

	nv show interface <interface-id> counters nvl drops Displays NVLink interface drop counters (e.g., QP1 drops for receive and transmit).	
Syntax Description	interface-id	NVLink interface name (e.g., acp142)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface acp142 counters nvl drops Counter Receive Transmit --------- ------- -------- qp1-drops 0 0	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/counters/nvl/drops	

Related Commands	nv show interface counters nvl
Notes	

5.10.5.17 nv show interface ipv4

	nv show interface <interface-id> ipv4 {address | dhcp-client | gateway} Displays IPv4 configuration and state of an IPoIB/eth0 interface.	
Syntax Description	interface-id	Name of the interface to display (e.g., ib0, eth0)
	address	Display the IPv4 address configuration of an IPoIB/eth0 interface
	dhcp-client	Display the DHCPv4 configuration and state of an IPoIB/eth0 interface
	gateway	Display the IPv4 gateway configuration of an IPoIB/eth0 interface
Default	N/A	
History	25.02.70xx	

Example	

```
admin@nvos:~$ nv show interface eth0 ipv4
                operational                          applied
--------------  -----------------------------------  -------
dhcp-client
  state                     enabled                    enabled
  set-hostname              enabled                    enabled
  lease
    fixed-address           10.7.148.195
    subnet-mask             255.255.248.0
    routers                 10.7.144.1
    broadcast-address       10.7.151.255
    filename                /uefiboot/bootx64-signed.efi
    server-name             10.7.77.122
    dhcp-lease-time         21600
    dhcp-message-type       5
    host-name               juliet-195
    domain-name             mtl.labs.mlnx
    renew                   2025-09-09 05:17:56
    rebind                  2025-09-09 07:49:37
    expire                  2025-09-09 08:34:37
    [domain-search]         lab.nvidia.com.
    [domain-search]         mlnx.
    [domain-search]         mtl.labs.mlnx.
    [domain-search]         mtr.labs.mlnx.
    [domain-search]         nbulabs.nvidia.com.
    [domain-search]         nvidia.com.
    [domain-name-servers]   10.7.77.135
    [domain-name-servers]   10.7.77.192
    [domain-name-servers]   10.198.0.169
    [domain-name-servers]   10.211.0.124
    [ntp-servers]           10.7.77.134
    [ntp-servers]           10.7.77.135
[address]                   10.7.148.195/21
[gateway]                   10.7.144.1
arp-timeout                 1800                       1800
```

```
admin@nvos:~$ nv show interface eth0 ipv4 address

-------------------------------------
10.7.148.195/21
```

```
admin@nvos:~$ nv show interface eth0 ipv4 dhcp-client
              operational  applied
------------  -----------  -------
state                     enabled                    enabled
set-hostname              enabled                    enabled
lease
  fixed-address           10.7.148.195
  subnet-mask             255.255.248.0
  routers                 10.7.144.1
  broadcast-address       10.7.151.255
  filename                /uefiboot/bootx64-signed.efi
  server-name             10.7.77.122
  dhcp-lease-time         21600
  dhcp-message-type       5
  host-name               juliet-195
  domain-name             mtl.labs.mlnx
  renew                   2025-09-09 07:35:23
  rebind                  2025-09-09 10:32:56
  expire                  2025-09-09 11:17:56
  [domain-search]         lab.nvidia.com.
  [domain-search]         mlnx.
  [domain-search]         mtl.labs.mlnx.
  [domain-search]         mtr.labs.mlnx.
  [domain-search]         nbulabs.nvidia.com.
  [domain-search]         nvidia.com.
  [domain-name-servers]   10.7.77.135
  [domain-name-servers]   10.7.77.192
  [domain-name-servers]   10.198.0.169
  [domain-name-servers]   10.211.0.124
  [ntp-servers]           10.7.77.134
  [ntp-servers]           10.7.77.135
```

REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/ipv4
Related Commands	nv show interface nv set interface ipv4 nv unset interface ipv4
Notes	

5.10.5.18 nv show interface ipv6

	nv show interface <interface-id> ipv6 {address | dhcp-client | gateway} Displays IPv6 configuration and state of an IPoIB/eth0 interface.	
Syntax Description	interface-id	Name of the interface to display (e.g., ib0, eth0)
	address	Display the IPv6 address configuration of an IPoIB/eth0 interface
	dhcp-client	Display the DHCPv6 configuration and state of an IPoIB/eth0 interface
	gateway	Display the IPv6 gateway configuration of an IPoIB/eth0 interface
Default	N/A	
History	25.02.70xx	
Example		

```
admin@nvos:~$ nv show interface eth0 ipv6
                operational                            applied
--------------  -------------------------------------  -------
dhcp-client
  state         enabled                                enabled
  set-hostname  enabled                                enabled
  lease
    client-id   0:3:0:1:e0:9d:73:c:5d:8c
    server-id   0:1:0:1:25:f2:90:34:0:50:56:8b:e4:f9
    [ia-na]     73:0c:5d:8c
[address]       fdfd:fdfd:7:145::1000:4751/128
[address]       fdfd:fdfd:7:145:e29d:73ff:fe0c:5d8c/64
[gateway]
autoconf        enabled                                enabled
```

```
admin@nvos:~$ nv show interface eth0 ipv6 address

------------------------------------
fdfd:fdfd:7:145::1000:4751/128
fdfd:fdfd:7:145:e29d:73ff:fe0c:5d8c/64
```

```
admin@nvos:~$ nv show interface eth0 ipv6 dhcp-client
                operational  applied
--------------  -----------  -------
state           enabled                             enabled
set-hostname    enabled                             enabled
lease
  client-id     0:3:0:1:e0:9d:73:c:5d:8c
  server-id     0:1:0:1:25:f2:90:34:0:50:56:8b:e4:f9
  [ia-na]       73:0c:5d:8c
```

REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/ipv6
Related Commands	nv show interface nv set interface ipv6 nv unset interface ipv6
Notes	

5.10.5.19 nv show interface ipv4 dhcp-client lease

	nv show interface <interface-id> ipv4 dhcp-client lease Displays the DHCPv4 client lease details for a specific interface	
Syntax Description	interface-id	Name of the interface to display (e.g., ib0, eth0)
Default	N/A	
History	25.02.70xx	

Example	admin@nvos:~$ nv show interface eth0 ipv4 dhcp-client lease operational applied fixed-address 10.7.148.195 subnet-mask 255.255.248.0 routers 10.7.144.1 broadcast-address 10.7.151.255 filename /uefiboot/bootx64-signed.efi server-name 10.7.77.122 dhcp-lease-time 21600 dhcp-message-type 5 host-name juliet-195 domain-name mtl.labs.mlnx renew 2025-09-09 07:35:23 rebind 2025-09-09 10:32:56 expire 2025-09-09 11:17:56 [domain-search] lab.nvidia.com. [domain-search] mlnx. [domain-search] mtl.labs.mlnx. [domain-search] mtr.labs.mlnx. [domain-search] nbulabs.nvidia.com. [domain-search] nvidia.com. [domain-name-servers] 10.7.77.135 [domain-name-servers] 10.7.77.192 [domain-name-servers] 10.198.0.169 [domain-name-servers] 10.211.0.124 [ntp-servers] 10.7.77.134 [ntp-servers] 10.7.77.135
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/ipv4
Related Commands	nv show interface nv set interface ipv4 dhcp-client state enabled nv unset interface ipv4
Notes	

5.10.5.20 nv show interface ipv6 dhcp-client lease

	nv show interface <interface-id> ipv6 dhcp-client lease Displays the DHCPv6 client lease details for a specific interface	
Syntax Description	interface-id	Name of the interface to display (e.g., ib0, eth0)
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv show interface eth0 ipv6 dhcp-client lease operational applied client-id 0:3:0:1:e0:9d:73:c:5d:8c server-id 0:1:0:1:25:f2:90:34:0:50:56:8b:e4:f9 ia-na ======== IA-NA Starts Renew Rebind IA Addr IA Addr Starts Preferred Life Max Life 73:0c:5d:8c 2025-09-09 04:34:42 2025-09-09 05:34:42 2025-09-09 06:34:42 fdfd:fdfd:7:145::1000:4751 2025-09-09 04:34:42 2025-09-09 05:12:12 2025-09-09 05:34:42 admin@juliet-195-mgmt2:~$	
REST API	GET https://<ip>/nvue_v1/interface/<interface-id>/ipv6	
Related Commands	nv show interface nv set interface ipv6 dhcp-client state enabled nv unset interface ipv6	
Notes		

5.10.5.21 nv set/unset interface link state

	nv set interface <interface-id> link state {value} nv unset interface <interface-id> link state {value} Set/unset the administrative link state of a given IPoIB/eth0 interface.	
Syntax Description	interface-id	Name of the interface whose link state to set (e.g., eth0)
	value	New value for the link state: up, down
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv unset interface ib0 link state admin@nvos:~$ nv set interface ib0 link state up admin@nvos:~$ nv unset interface eth0 link state admin@nvos:~$ nv set interface eth0 link state up	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/link/state	
Related Commands	nv show interface nv set interface link nv unset interface link state	
Notes		

5.10.5.22 nv set/unset interface description

	nv set interface <interface-id> description nv unset interface <interface-id> description Sets the description of a given IPoIB/eth0 interface. The unset form of the command sets the description of a given IPoIB/eth0 interface to empty.	
Syntax Description	interface-id	Name of the interface whose link state to set, (e.g. ib0, eth0)
	value	New value for the description.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface ib0 description "IPoIB interface" admin@nvos:~$ nv unset interface ib0 description admin@nvos:~$ nv set interface eth0 description "mgmt interface" admin@nvos:~$ nv unset interface eth0 description	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>	
Related Commands	nv show interface nv unset interface description	
Notes		

5.10.5.23 nv set/unset interface ipv4 address

	nv set interface <interface-id> ipv4 address <ipv4-prefix-id> nv unset interface <interface-id> ipv4 address <ipv4-prefix-id> Sets the IPv4 address of a given IPoIB/eth0 interface. The unset form of the command deletes one or more IPv4 addresses assigned to a given IPoIB/eth0 interface.	
Syntax Description	interface-id	Name of the interface whose IPv4 address to set (e.g., ib0, eth0)
	ipv4-prefix-id	IPv4 address and netmask to assign to the interface
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv set interface ib0 ipv4 address 10.10.1.1/8 admin@nvos:~$ nv set interface eth0 ipv4 address 10.10.1.1/8	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv4/address	
Related Commands	nv show interface nv unset interface ipv4	
Notes		

5.10.5.24 nv set/unset interface ipv6 address

	nv set interface <interface-id> ipv6 address <ipv6-prefix-id> nv unset interface <interface-id> ipv6 address <ipv6-prefix-id> Sets the IPv6 address of a given IPoIB/eth0 interface. The unset form of the command deletes one or more IPv6 addresses assigned to a given IPoIB/eth0 interface.	
Syntax Description	interface-id	Name of the interface whose IPv6 address to set (e.g., ib0, eth0)
	ipv6-prefix-id	IPv6 address and netmask to assign to the interface
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv set interface ib0 ipv6 address fdfd:fdfd:7:145::1000:4751/128 admin@nvos:~$ nv set interface eth0 ipv6 address fdfd:fdfd:7:145::1000:4751/128	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv6/address	
Related Commands	nv show interface nv unset interface ipv6	
Notes		

5.10.5.25 nv unset interface

	nv unset interface <interface-id> Sets all attributes of an IPoIB/eth0 interface to default values.	
Syntax Description	interface-id	Name of the interface to set to default values (e.g., eth0)
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv unset interface eth0`	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>	
Related Commands	nv show interface nv set interface	
Notes		

5.10.5.26 nv set/unset interface link mtu

	nv set interface <interface-id> link mtu <bytes> nv unset interface <interface-id> link mtu Sets the Maximum Transmission Unit (MTU) of this interface. The unset form of the command resets the MTU to its default.	
Syntax Description	interface-id	Name of the interface to display (e.g., eth0)
	bytes	Range: 1280–9000
Default	1500	
History	25.02.2002	
Example	`admin@nvos:~$ nv set interface eth0 link mtu 1500`	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/link	
Related Commands	nv show interface link	
Notes		

5.10.5.27 nv set/unset interface link speed

	nv set interface <interface-id> link speed <speed> nv unset interface <interface-id> link speed Sets the interface speed. The unset form of the command resets the speed setting for this interface to its default value.	
Syntax Description	interface-id	Name of the interface to display (e.g., eth0)
	speed	10M, 100M, 1000M
Default	1000M	

History	25.02.2002
Example	`admin@nvos:~$ nv set interface eth0 link speed 100M`
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/link
Related Commands	nv show interface link
Notes	This command is currently not supported in IPoIB interfaces.

5.10.5.28 nv set/unset interface link duplex

	nv set interface <interface-id> link duplex <duplex> nv unset interface <interface-id> link duplex Sets the interface duplex. The unset form of the command resets the duplex setting for this interface to its default value.	
Syntax Description	interface-id	The interface (e.g., eth0)
	duplex	full, half
Default	full	
History	25.02.2002	
Example	`admin@nvos:~$ nv set interface eth0 link duplex full`	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/link	
Related Commands	nv show interface link	
Notes	This command is currently not supported in IPoIB interfaces.	

5.10.5.29 nv set interface link auto-negotiate

	nv set interface <interface-id> link auto-negotiate <auto> Sets link speed and characteristic auto negotiation.	
Syntax Description	interface-id	The interface (e.g., eth0)
	auto	enabled, disabled
Default	enabled	
History	25.02.2002 25.02.70xx `<auto>` is now `enabled | disabled` instead of `on | off`	
Example	`admin@nvos:~$ nv set interface eth0 link auto-negotiate enabled`	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/link	
Related Commands	nv show interface link	
Notes	This command is currently not supported in IPoIB interfaces.	

5.10.5.30 nv set interface ipv6 autoconf

	nv set interface <interface-id> ipv6 autoconf <autoconf> IPv6 Stateless Address Autoconfiguration (SLAAC)	
Syntax Description	interface-id	The interface
	autoconf	enable, disable
Default	Enable	
History	25.02.2002	
Example	`admin@nvos:~$ nv set interface eth0 ipv6 autoconf enable`	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv6	
Related Commands	nv unset interface ipv6 gateway	
Notes		

5.10.5.31 nv set/unset interface ipv4 gateway

	nv set interface <interface-id> ipv4 gateway {gateway-ip} nv unset interface <interface-id> ipv4 gateway {gateway-ip} Sets default gateway IPv4 address for an interface. The unset form of the command unsets default gateway IPv4 address for an interface.	
Syntax Description	interface-id	The interface
	gateway-ip	The gateway IPv4 address
Default	N/A	
History	25.02.70xx	
Example	`admin@nvos:~$ nv set interface eth0 ipv4 gateway 1.1.1.1` `admin@nvos:~$ nv unset interface eth0 ipv4 gateway 1.1.1.1`	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv4/gateway	
Related Commands	nv show interface ipv4	
Notes		

5.10.5.32 nv set/unset interface ipv6 gateway

	nv set interface <interface-id> ipv6 gateway {gateway-ip} nv unset interface <interface-id> ipv6 gateway {gateway-ip} Sets default gateway IPv6 address for an interface. The unset form of the command unsets default gateway IPv6 address for an interface.	
Syntax Description	interface-id	The interface

	gateway-ip	The gateway IPv6 address
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv set interface eth0 ipv6 gateway fe80::1 admin@nvos:~$ nv unset interface eth0 ipv6 gateway fe80::1	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv6/gateway	
Related Commands	nv show interface ipv6	
Notes		

5.10.5.33 nv set/unset interface ipv4 arp-timeout

	nv set interface <interface-id> ipv4 arp-timeout <time> nv unset interface <interface-id> ipv4 arp-timeout Sets IPv4 ARP timeout (in seconds) for an interface. The unset form of the command unsets IPv4 ARP timeout (in seconds) for an interface.	
Syntax Description	interface-id	The interface
	time	Seconds. Range: 60–28800
Default	1800	
History	25.02.70xx	
Example	admin@nvos:~$ nv set interface eth0 ipv4 arp-timeout 2100 admin@nvos:~$ nv unset interface eth0 ipv4 arp-timeout	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv4	
Related Commands		
Notes		

5.10.5.34 nv set/unset interface link breakout

	nv (un)set interface <interface-id> link breakout [mode] Set/unset interface breakout mode.	
Syntax Description	interface-id	The interface for which to set breakout.
	mode	The breakout mode to set.
Default	Disabled	
History	25.02.2002	

Example	admin@nvos:~$ nv set interface sw1p1 link breakout 2x-ndr admin@nvos:~$ nv unset interface sw1p1 link breakout
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ip
Related Commands	nv action change system profile breakout-mode enabled
Notes	• Breakout configuration may only be applied when system profile allows it (i.e., breakout-mode is enabled). • To get the current system profile, run: nv show system profile • To enable breakout-mode in system profile, run: nv action change system profile breakout-mode enabled ...

5.10.5.35 VRF

5.10.5.35.1 nv show vrf

	show vrf {vrf-id}{loopback | loopback ip | loopback ip address {ip-prefix-id}} Shows VRFs.	
Syntax Description	vrf-id	VRF
	loopback	Return loopback interface details of a VRF.
	loopback ip	Return IP details of a VRF loopback interface.
	loopback ip address	Return details of the IP addresses.
	ip-prefix-id	IPv4 or IPv6 address and route prefix in CIDR notation
	<none>	Shows all VRFs
Default	N/A	
Configuration Mode	config	
History	25.02.2002	

Example	admin@nvos:~$ nv show vrf Name Table Summary --------- ----- ---------- + default 254 IP Address: 127.0.0.1/8 default IP Address: ::1/128 + mgmt 1001 IP Address: 127.0.0.1/8 mgmt IP Address: ::1/128 admin@nvos:~$ nv show vrf mgmt operational applied description ---------- ---------- ---------- table 1001 auto The routing table number, between 1001-1255, used by the named VRF.... evpn enable off Turn the feature 'on' or 'off'. The default is 'off'. loopback ip [address] 127.0.0.1/8 127.0.0.1/8 static IPv4 or IPv6 address [address] ::1/128 ::1/128 ptp enable on Turn the feature 'on' or 'off'. The default is 'on'. router bgp enable off Turn the feature 'on' or 'off'. The default is 'off'. ospf enable off Turn the feature 'on' or 'off'. The default is 'off'. [rib] RIB Routes [static] Routes admin@nvos:~$ nv show vrf mgmt loopback operational applied description ---------- ---------- ---------- ---------- ip [address] 127.0.0.1/8 127.0.0.1/8 static IPv4 or IPv6 address [address] ::1/128 ::1/128 admin@nvos:~$ nv show vrf mgmt loopback ip operational applied description --------- ---------- ---------- ---------- [address] 127.0.0.1/8 127.0.0.1/8 static IPv4 or IPv6 address [address] ::1/128 ::1/128 admin@nvos:~$ nv show vrf mgmt loopback ip address ---------- 127.0.0.1/8 ::1/128 admin@nvos:~$ nv show vrf mgmt loopback ip address 127.0.0.1/8 operational applied description -- ---------- ------- ----------
REST API	GET https://<ip>/nvue_v1/vrf/{vrf-id}
Related Commands	
Notes	FRR must be running to be able to use this command.

5.10.5.35.2 nv set/unset interface vrf

	nv set interface <interface-id> vrf {vrf-name} nv unset interface <interface-id> vrf Assigns an interface to a VRF. The unset form of the command unsets an interface to a VRF.

Syntax Description	interface-id	The interface
	vrf-name	The VRF: default/ mgmt
Default	N/A	
History	25.02.70xx	
Example	admin@nvos:~$ nv set interface eth0 vrf mgmt admin@nvos:~$ nv unset interface eth0 vrf	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>	
Related Commands	nv unset interface eth0 vrf	
Notes		

5.10.5.36 IP DHCP Client

5.10.5.36.1 nv set/unset interface ipv4 dhcp-client state

	nv set interface <interface-id> ipv4 dhcp-client state {enabled | disabled} nv unset interface <interface-id> ipv4 dhcp-client Enables/disables DHCP client for an interface. The unset form of the command returns the DHCP client to its default state.	
Syntax Description	interface-id	The interface
Default	Enabled	
History	25.02.70xx	
Example	admin@nvos:~$ nv set interface eth0 ipv4 dhcp-client state enabled	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv4/dhcp-client	
Related Commands	nv set interface ipv6 dhcp-client state	
Notes	Configuring the DHCP client for either IPv4 or IPv6 will set the same configuration for both.	

5.10.5.36.2 nv set/unset interface ipv4 dhcp-client set-hostname

	nv set interface <interface-id> ipv4 dhcp-client set-hostname {enabled | disabled} nv unset interface <interface-id> ipv4 dhcp-client set-hostname Allows/disallows DHCP client to set system hostname from DHCP. The unset form of the command returns the system hostname to its default state.	
Syntax Description	interface-id	The interface
Default	Enabled	

History	25.02.70xx
Example	admin@nvos:~$ nv set interface eth0 ipv4 dhcp-client set-hostname enabled
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv4/dhcp-client
Related Commands	nv set interface ipv6 dhcp-client set-hostname nv set interface ipv4 dhcp-client state
Notes	Configuring the DHCP client for either IPv4 or IPv6 will set the same configuration for both.

5.10.5.36.3 nv set/unset interface ipv6 dhcp-client state

	nv set interface <interface-id> ipv6 dhcp-client state {enabled | disabled} nv unset interface <interface-id> ipv6 dhcp-client Enables/disables DHCPv6 client for an interface. The unset form of the command returns the DHCPv6 client to its original state.	
Syntax Description	interface-id	The interface
Default	Enabled	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface eth0 ipv6 dhcp-client state enabled	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv6/dhcp-client	
Related Commands	nv set interface ipv4 dhcp-client state	
Notes	Configuring the DHCP client for either IPv4 or IPv6 will set the same configuration for both.	

5.10.5.36.4 nv set interface ipv6 dhcp-client set-hostname

	nv set interface <interface-id> ipv6 dhcp-client set-hostname {enabled | disabled} nv unset interface <interface-id> ipv6 dhcp-client set-hostname Allows/disallows DHCP client to set system hostname from DHCPv6. The unset form of the command returns the DHCP client to its default state.	
Syntax Description	interface-id	The interface
Default	Enabled	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface eth0 ipv6 dhcp-client set-hostname enabled	
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/ipv6/dhcp-client	
Related Commands	nv set interface ipv4 dhcp-client set-hostname nv set interface ipv6 dhcp-client state	

Notes	Configuring the DHCP client for either IPv4 or IPv6 will set the same configuration for both.

5.10.5.36.5 nv action renew interface ipv4 dhcp-client

	nv action renew interface <interface-id> ipv4 dhcp-client Renews DHCPv4 lease for this interface.	
Syntax Description	interface-id	The interface
Default	N/A	
History	25.02.70xx	
Example	`admin@nvos:~$ nv action renew interface eth0 ipv4 dhcp-client`	
REST API	POST https://<ip>/nvue_v1/interface/<interface-id>/ipv4	
Related Commands		
Notes		

5.10.5.36.6 nv action renew interface ipv6 dhcp-client

	nv action renew interface <interface-id> ipv6 dhcp-client Renews DHCPv6 lease for this interface.	
Syntax Description	interface-id	The interface
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv action renew interface eth0 ipv6 dhcp-client`	
REST API	POST https://<ip>/nvue_v1/interface/<interface-id>/ipv6/dhcp-client	
Related Commands		
Notes		

5.11 Profile

NVOS enables users to manage the system profile. The profile includes settings such as adaptive routing state, group numbers, breakout mode state, and the number of SWIDs.

The user is permitted to manage the number of adaptive-routing groups. For instance:

```
admin@nvos:~$ nv action change system profile adaptive-routing-groups 2048
The operation will reset the system configuration and initiate a reboot.
Type [y] to confirm the system profile change and reboot.
```

```
Type [N] to abort.
```

Performing profile change will lead to factory reset of the system with preserving basic configurations, for more information please see **keep basic** in Restoring Factory Default Configuration section.

5.11.1 Profile Commands

- Profile Commands

5.11.2 Profile Commands

5.11.2.1 nv show system profile

	nv show system profile Displays system profile.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show system profile operational applied ---------- ----------- -------- adaptive-routing enabled adaptive-routing-groups 1792 breakout-mode enabled ib-routing disabled num-of-swids 1
REST API	GET https://<ip>/nvue_v1/system/profile
Related commands	nv action change system profile
Notes	

5.11.2.2 nv action change system profile

	nv action change system profile [adaptive-routing-groups <ar_groups>] [force] Sets the profile of the system.	
Syntax Description	adaptive-routing-groups	Sets adaptive routing groups. Range: 128-1792 (must be multiples of 128). Allowed only when adaptive routing is enabled.
	force	Force the action without asking for user confirmation.
Default	adaptive-routing: enabled adaptive-routing-groups: 1792	
History	25.02.2002	

Example	admin@nvos:~$ nv action change system profile adaptive-routing-groups 1792 The operation will reset the system configuration and initiate a reboot. Type [y] to confirm the system profile change and reboot. Type [N] to abort.
REST API	POST https://<ip>/nvue_v1/system/profile
Related Commands	nv show system profile nv set/unset interface link breakout
Notes	This operation will perform reset factory with preserving basic configurations

5.12 Resource Management

NVOS provides commands for managing and monitoring essential system resources. These tools provide general information about the system, insights into the system's overall health, and resources utilization and usage.

5.12.1 Resource Management Commands

5.12.2 Resource Management Commands

5.12.2.1 nv show system

	nv show system Show general system information.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.4002 Updated memory field in output 25.02.5002 Updated output: removed "memory" and "build" fields and added "product-release" field

Example	admin@nvos:~$ nv show system operational applied ---------- ---------- ---------- uptime 2:32:29 hostname switch-85-mgmt2 product-name nvos product-release 25.02.4207 status System is ready contact Veni Veni location Asia Asia date-time local-time 2025-05-19 11:38:35 timezone Asia/Jerusalem Asia/Jerusalem health status OK version product-release 25.02.4207
REST API	GET https://<ip>/nvue_v1/system
Related commands	
Notes	

5.12.2.2 nv show system cpu

	nv show system cpu Show system CPU.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.4002—Updated output
Example	admin@nvos:~$ nv show system cpu operational ---------- ---------- model Intel(R) Core(TM) i3-8100H CPU @ 3.00GHz core-count 4 total-utilization 39.9% load-average one-minute 1.67 five-minute 1.96 fifteen-minute 1.87 Cores ======== CPU Utilization ---- ---------- CPU0 38.5% CPU1 43.4% CPU2 39.5% CPU3 38.2%
REST API	GET https://<ip>/nvue_v1/system/cpu
Related Commands	
Notes	

5.12.2.3 nv show system memory

	nv show system memory Show system memory.
Syntax Description	N/A
Default	N/A

History	25.02.2002
Example	admin@nvos:~$ nv show system memory B - Bytes, KB - Kilobytes, MB - Megabytes, GB - Gigabytes, % - Percent Physical buffer: 53.12 MB Physical cache: 1.82 GB Physical free: 5.74 GB Physical total: 7.46 GB Physical used: 1.72 GB Physical utilization: 23.06 % Swap free: 0 B Swap total: 0 B Swap used: 0 B Swap utilization: 0.00 %
REST API	GET https://<ip>/nvue_v1/system/memory
Related Commands	
Notes	

5.12.2.4 nv show system disk

	nv show system disk Get system disk information.
Syntax Description	N/A
Default	N/A
History	25.02.3000
Example	admin@nvos:~$ nv show system disk usage ======== Mount Point Filesystem Size Used Avail Use% ----------- ---------- ---------- ---------- ---------- ----- /host /dev/sda3 146561.67M 12001.51M 134560.16M 8.20% /var/log /dev/loop1 14660.34M 65.66M 14594.69M 0.50%
REST API	GET https://<ip>/nvue_v1/system/disk
Related Commands	nv show system disk usage
Notes	

5.12.2.5 nv show system disk usage

	nv show system disk usage Get system storage utilization information.
Syntax Description	N/A
Default	N/A
History	25.02.3000
Example	admin@nvos:~$ nv show system disk usage Mount Point Filesystem Size Used Avail Use% ----------- ---------- --------- -------- -------- ------ /host /dev/sda3 14549.41M 4611.09M 9938.31M 31.70% /var/log /dev/loop1 3951.72M 190.77M 3760.95M 5.10%

REST API	GET https://<ip>/nvue_v1/system/disk/usage
Related Commands	nv show system disk
Notes	

5.12.2.6 nv action erase system disk

	nv action erase system disk [force] Wipe all the SSD data.	
Syntax Description	force	Force the action without asking for user confirmation (the warning message indicating that the disk data will be unrecoverable will be bypassed).
Default	N/A	
History	25.02.3000	
Example	`admin@nvos:~$ nv action erase system disk` `admin@nvos:~$ nv action erase system disk force`	
REST API	GET https://<ip>/nvue_v1/system/disk	
Related Commands	nv action change system security sed-password 12345678	
Notes	The procedure is secure by doing crypto erase and block erase, ensuring the data is unrecoverable. ⚠ To reboot after the SSD wipe action finishes, use the following command **/bin/reboot** For more details, see in the Security section.	

5.13 Security

5.13.1 SSD Wipe

The SSD wipe command below performs a Secure Erase and Crypto Erase.

- Cryptographic Erase (CE) is a secure method for sanitizing entire drives and storage devices by deleting the encryption keys used to protect the data.
- Secure erasure is the process of permanently and irreversibly removing data from a storage device, ensuring that the data cannot be recovered using any means, even with advanced data recovery tools.

5.13.1.1 Prerequisites

Before performing the SSD wipe, complete the following requirements:

- Record the currently installed NVOS version.
- Verify that all system component versions are aligned with the installed release.
- Ensure that the same NVOS version is available for the recovery procedure.
- Back up any required system configuration or data.
- Establish and verify a working serial console connection.

> ⚠ After the SSD wipe and subsequent reboot, no operating system will be installed on the system. NVOS SSH access will no longer be available, and the recovery procedure must be performed through the serial console.

5.13.1.1.1 Serial Console Connection Using SoL

On GB200 and newer NVLink switches with BMC Serial over LAN support, the host serial console can be accessed using SoL (Serial over LAN).

SoL provides access to the host CPU console through an SSH connection to TCP port `2200` on the BMC.

SoL is implemented using the open-source `openbmc/obmc-console` project, which provides the OpenBMC host-console infrastructure.

5.13.1.1.2 Enable SoL

SoL is disabled by default.

1. Connect to the BMC:

```
ssh root@<bmc-ip>
```

2. Enter the BMC password.
3. Start the SoL services:

```
systemctl start obmc-console-ssh.socket
systemctl start obmc-console@ttyS2.service
```

4. Connect to the host CPU console:

```
ssh -p 2200 root@<bmc-ip>
```

5. Enter the BMC password.

> While SoL is enabled, console output remains visible through the physical serial port, but input cannot be entered through the physical serial connection.

5.13.1.1.3 Verify the SoL Connection

Verify that the serial console connection is working before performing the SSD wipe.

From an NVOS SSH session, run:

```
who
```

Example output:

```
admin    pts/0        2026-07-28 10:58 (.1.2.3.4)
admin    ttyS0        2026-07-28 11:07
```

Confirm that a serial TTY session appears in the output.

To verify that console output is visible through SoL, keep the BMC port `2200` session open. From a separate NVOS SSH session, run:

```
sudo sh -c 'echo HELLO-FROM-SSH > /dev/console'
```

Confirm that the following message appears in the SoL session:

```
HELLO-FROM-SSH
```

> **Warning:** Do not proceed with the SSD wipe unless the serial console connection has been successfully verified.

5.13.1.1.4 SoL Service Notes

The `obmc-console` service must be started only after the BMC-ready script has completed. This prevents the BMC-ready script from changing the UART selection in the CPLD after a SoL session has been initiated.

The `obmc-console` server and the `serial-getty` service both use `/dev/ttyS2`. Therefore:

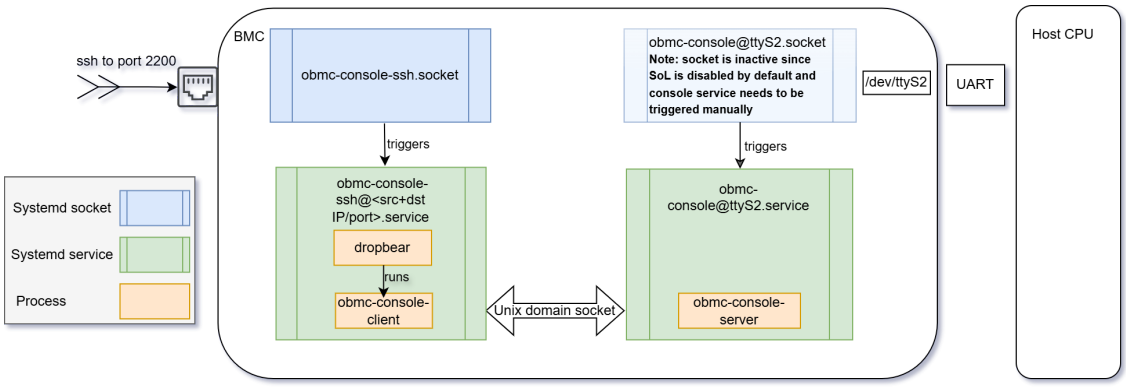
- `obmc-console` conflicts with `serial-getty`.
- Enabling `obmc-console` disables `serial-getty`.
- `serial-getty` remains disabled while `obmc-console` is enabled, even when no SoL SSH sessions are active.
- SoL is disabled following a BMC reboot and must be enabled again when required.

5.13.1.2 Perform Disk Wipe

> ⚠ Before initiating this procedure, make sure to note the currently installed NVOS version and verify that all system component versions are aligned with the release.

1. Pre-Required is to be connected via serial.
   After the Disk Wipe procedure and reboot, there will be no OS installed, and you will need to proceed with the Recovery procedure detailed in the section below. So, the requirement is to have a connection via serial.
   Serial connection:
   In GB200 and newer, there is the ability from the BMC device to see the serial connection by using the feature name:
   SoL - Serial Over LAN
   SoL allows the user to access the host CPU console by connecting to port 2200 via SSH to the BMC.
   SoL is implemented by the open-source project: [openbmc/obmc-console: OpenBMC host console infrastructure]

   SoL module view:



   How to connect via serial:

   a. Enable the services from the BMC (by default the SoL is disabled)
      Connect to BMC:

      ```
      ssh root@<bmc-ip>
      ```

      Enter BMC password.

   b. Enable the SoL feature

   c.
      ```
      systemctl start obmc-console-ssh.socket
      systemctl start obmc-console@ttyS2.service
      ```

   d. Connect to the CPU console:

      ```
      ssh -p 2200 root@<bmc ip>
      ```

      Enter BMC password.

Note: User will still see console output on the physical serial port but will not be able to enter any input.

e. To assert that this serial connection is indeed working before proceeding with the SSD Wipe is expected to see a tty node.
From NVOS SSH connection

```
admin@juliet-XXX:~$ who
admin    pts/0        2026-07-28 10:58 (.1.2.3.4)
admin    ttyS0        2026-07-28 11:07
```

f. In addition, from the NVOS SSH connection, you can write to the console and see that the serial is showing the inputs,
From one terminal connected to NVOS via SSH, type the cmd below, and from the BMC 2200 port connection you will see the msg (if the connection is indeed working)
Example:

```
sudo sh -c 'echo HELLO-FROM-SSH > /dev/console'
```

g. In order to disconnect SoL sessions and disable obmc console service from the BMC Linux shell do:

```
systemctl stop obmc-console@ttyS2.service
systemctl stop obmc-console-ssh.socket
systemctl start serial-getty@ttyS2.service
```

h. Notes:

i. Persistency
Following a BMC reboot SoL service shall be disabled.

ii. Service dependencies

- obmc-console service should start only once bmc ready script has been completed in order to avoid a situation where bmc ready scripts changes uart selection in CPLD after SoL session has been initiated.
- getty and obmc console server are both using /dev/ttyS2.
Therefore:
  - obmc-console service conflicts with serial-getty service.
  - getty will be disabled as soon as obmc console server is enabled and shall remain disabled as long as console server is enabled even when no ssh sessions are active.

2. To perform the SSD wipe action, run the following command:

```
admin@nvos:~$ nv action erase system disk
```

More details can be found in the Resource Management Commands section.

> ⚠ Note that once the action is completed, it will be impossible to recover the SSD data. Additionally, the user will be able to connect to the system until the next reboot, as the file system (only mandatory directories) is loaded into RAM. After a reboot, the user will not be able to see any SSD partitions.

3. Reboot after the SSD wipe action finishes, use the following command:

```
/sbin/reboot
```

5.13.1.3 Persistency

Following a BMC reboot SoL service shall be disabled.

5.13.2 Recovery Flow After SSD Wipe

5.13.2.1 Prerequisites

Before starting the recovery process, ensure the following requirements are met:

> ⚠ During recovery, use the same NVOS version that was previously installed to avoid version incompatibility with other platform components.

- PXE Server Setup
  - A PXE (Preboot Execution Environment) server must be installed and running in the lab network
  - The PXE server should be configured to point to the NVIDIA ONIE image and automatically boot from it without user interaction
- Required Resources
  - NVIDIA ONIE image
  - NVOS image
  - Provisioning package

> ⚠ Performing an SSD wipe will erase the previous NVOS configuration, and it will not be recoverable. Ensure that any required configurations or data are backed up before proceeding.

5.13.2.2 Recovery Steps

1. Wait for NVIDIA ONIE installation to finish (should take up to 3 minutes).
2. Connect to ONIE via ssh. Please refer to UM for ONIE default credentials.
3. Once connected to ONIE, stop onie install by running `onie-stop`.
4. Copy provisioning package (e.g., sed_provisioning_83.03.0001.tgz) to `/tmp`.
5. Extract provisioning script and run it.

```
cd /tmp
tar -xzf sed_provisioning_83.03.0001.tgz
./sedutil_init.sh
```

6. Wait for the script to finish (which can take up to 2 minutes). The device will perform power-cycle.
7. Perform a new NVOS image install following the guidelines in the Installing a New NVOS Image section.

5.13.3 Security Commands

- Password Hardening
- Security Commands
- SED Commands

5.13.4 Password Hardening

NVOS includes a robust password hardening feature to enhance security and protect user accounts.

By default, this feature is disabled.

If enabled, the passwords must meet the following requirements:

- Minimum length of 8 characters
- Inclusion of at least the following:
  - One uppercase letter
  - One lowercase letter
  - One number
  - One special character from the set `` `~!@#$%^&*()-_+=|[{}];:',<.>/? `` and white space
- The password cannot reuse any of the last 10 previously saved passwords
- The password cannot contain the username

The user can choose to enable or disable/change one or more policies or even deactivate the feature entirely.

> ⚠ Password hardening check are forced in NVUE during SET/PATCH operations. Any changes to the policies or feature state must be applied first and will only take effect for subsequent operations.

5.13.4.1 Password Hardening Commands

- Password Hardening Commands

5.13.4.2 Password Hardening Commands

- 5.13.4.2.1 nv show system security password-hardening
- 5.13.4.2.2 nv set system security password-hardening state
- 5.13.4.2.3 nv set system security password-hardening digits-class
- 5.13.4.2.4 nv set system security password-hardening expiration
- 5.13.4.2.5 nv set system security password-hardening expiration-warning
- 5.13.4.2.6 nv set system security password-hardening history-cnt
- 5.13.4.2.7 nv set system security password-hardening len-min
- 5.13.4.2.8 nv set system security password-hardening lower-class
- 5.13.4.2.9 nv set system security password-hardening reject-user-passw-match

5.13.4.2.1 nv show system security password-hardening

	nv show system security password-hardening Displays the password hardening rules applied on top of the switch.
Syntax Description	N/A
Default	The example contains the default values of the feature
History	25.02.2002
Example	admin@nvos:~$ nv show system security password-hardening operational applied ---------- ---------- ------- state enabled enabled reject-user-passw-match enabled enabled lower-class enabled enabled upper-class enabled enabled digits-class enabled enabled special-class enabled enabled expiration-warning 15 15 expiration 180 180 history-cnt 10 10 len-min 8 8
Rest API	GET https://<id>/nvue_v1/system/security/password_hardening
Related Commands	nv set system security password-hardening
Notes	Password hardening rules are applied only to locally stored passwords

5.13.4.2.2 nv set system security password-hardening state

	nv set system security password-hardening state <enabled | disabled> Enable or disable the password hardening feature	
Syntax Description	state	enabled, disabled
Default	disabled	
History	25.02.2002	
Example	admin@nvos:~$ nv set system security password-hardening state enabled	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/state/	
Related Commands	nv show system security password-hardening	
Notes	When password hardening is enabled - the switch does not accept hashed passwords	

5.13.4.2.3 nv set system security password-hardening digits-class

	nv set system security password-hardening digits-class<enabled | disabled> Enable or disable the requirement to enforce digits in the password

Syntax Description	digits-class	enabled, disabled
Default	enabled	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system security password-hardening state enabled`	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/digits-class/	
Related Commands	nv show system security password-hardening	
Notes		

5.13.4.2.4 nv set system security password-hardening expiration

	nv set system security password-hardening expiration [<integer days>] Number of days for password validity, afterwards user will be prompted to change his password	
Syntax Description	expiration	expiration days (1–365)
Default	180	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system security password-hardening expiration 200`	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/expiration/	
Related Commands	nv show system security password-hardening	
Notes		

5.13.4.2.5 nv set system security password-hardening expiration-warning

	nv set system security password-hardening expiration-warning [<integer days>] Number of days for password warning which will alert the user that he needs to change his password before it expires. The alert will appear on the login screen for the user	
Syntax Description	expiration-warning	warning days (1–30)
Default	15	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system security password-hardening expiration-warning 10`	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/expiration-warning/	
Related Commands	nv show system security password-hardening	

Notes	

5.13.4.2.6 nv set system security password-hardening history-cnt

	nv set system security password-hardening history-cnt [<integer count>] Number of passwords the system will compare the current password against. If the password is equal to one of the previously configured password, the system will reject it.	
Syntax Description	history-cnt	history count (1–100)
Default	10	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system security password-hardening history-cnt 5`	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/history-cnt/	
Related Commands	nv show system security password-hardening	
Notes		

5.13.4.2.7 nv set system security password-hardening len-min

	nv set system security password-hardening len-min [<integer length>] Set the minimum length for a password.	
Syntax Description	len-min	length (6–32)
Default	8	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system security password-hardening len-min 10`	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/len-min/	
Related Commands	nv show system security password-hardening	
Notes		

5.13.4.2.8 nv set system security password-hardening lower-class

	nv set system security password-hardening lower-class <enabled/disabled> Enable or disable the requirement to enforce lower case letters in the password.	
Syntax Description	lower-class	enabled/disabled
Default	enabled	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system security password-hardening lower-class <enabled/disabled>`	

REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/lower-class/
Related Commands	nv show system security password-hardening
Notes	

5.13.4.2.9 nv set system security password-hardening reject-user-passw-match

	nv set system security password-hardening reject-user-passw-match<enabled/disabled> Enable or disable allowing the username and password to be identical.	
Syntax Description	reject-user-passw-match	enabled/disabled
Default	enabled	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system security password-hardening state <enabled/disabled>`	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/reject-user-passw-match/	
Related Commands	nv show system security password-hardening	
Notes		

5.13.4.2.10 nv set system security password-hardening special-class

	nv set system security password-hardening special-class <enabled/disabled> Enable or disable the requirement to enforce special characters in the password.	
Syntax Description	special-class	feature state
Default	enabled	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system security password-hardening special-class disabled`	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/special-class/	
Related Commands	nv show system security password-hardening	
Notes	• The special characters allowed are: `~!@#$%^&*()-_+=\|[{}];:',<.>/? and white space • These characters must be accompanied by quotation marks "" in order to be received correctly in the password string. • Example: nv set system aaa user example password "123%$Aabcv"	

5.13.4.2.11 nv set system security password-hardening upper-class

	nv set system security password-hardening upper-class <enabled/disabled> Enable or disable the requirement to enforce upper case letters in the password.

Syntax Description	upper-class	enabled/disabled
Default	enabled	
History	25.02.2002	
Example	admin@nvos:~$ nv set system security password-hardening upper-class enabled	
REST API	PATCH https://<id>/nvue_v1/system/security/password_hardening/upper-class/	
Related Commands	nv show system security password-hardening	
Notes		

5.13.5 Security Commands

5.13.5.1 nv show system security gpu-mad-firewall

	nv show system security gpu-mad-firewall Display MAD firewall state.
Syntax Description	N/A
Default	enabled
History	25.03.0227
Example	admin@nvos:~$ nv show system security gpu-mad-firewall operational applied ---------- ---------- ---------- state enabled enabled
REST API	GET https://<ip>/nvue_v1/system/security/gpu-mad-firewall
Related Commands	
Notes	

5.13.5.2 nv set system security gpu-mad-firewall

	nv set system security gpu-mad-firewall state {state} Set MAD firewall state	
Syntax Description	state	enabled | disabled
Default	enabled	
History	25.03.0227	
Example	admin@nvos:~$ nv set system security gpu-mad-firewall state disabled created [rev_id: 1] admin@nvos:~$ nv config apply	
REST API	PATCH https://<ip>/nvue_v1/system/security/gpu-mad-firewall/state	
Related Commands		
Notes		

5.13.6 SED Commands

To support all SED features, a PBA version greater than 83.02.0003 is required.

5.13.6.1 nv action change system security sed-password

	nv action change system security [sed-password] Change the SED password by setting a new password chosen by the user.	
Syntax Description	sed-password	Minimum password length: 8 charachters Maximum password length: 250 charachters
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action change system security sed-password 12345678	
REST API	POST https://<ip>/nvue_v1/system/security	
Related Commands		
Notes	The password can be up to 250 characters but it is hashed down to 32 bytes to conform to the size required by the drive.	

5.14 System API

NVOS offers commands for managing and monitoring its external REST API interface. These capabilities allow users to configure, enable, and monitor API settings, ensuring secure and efficient interaction with external systems.

5.14.1 Key Functionalities

- API State Management: Enable or disable external REST API access.
- Port Configuration: Specify the port number on which the REST API listens, with a default setting of 443.
- Certificate Management: View and configure the CA certificate for mutual TLS (mTLS) connections. For more examples on how to import and set certificates, Please see "Certificates Management" section.
- Operational Metrics: Retrieve detailed API usage statistics, including active connections, total requests, and the status of ongoing requests.

5.14.2 System API Commands

5.14.3 System API Commands

5.14.3.1 nv show system api

	nv show system api Show NVUE external REST API configuration.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output

Example	<pre>admin@nvos:~$ nv show system api
	                    operational applied
	------------------ ----------- -----------
	state enabled enabled
	port 443 443
	certificate self-signed self-signed
	token-expiration 60 60
	[listening-address] any
	connections
	  active 1
	  accepted 1
	  handled 1
	  requests 1
	  reading 0
	  writing 1
	  waiting 0
	
	versions
	===========
	             operational
	    ------- -----------
	    nvue current
	    nvue_v1 current</pre>

REST API	GET https://<ip>/nvue_v1/system/api
Related Commands	nv show system api versions nv set/unset system api state nv set/unset system api port
Notes	

5.14.3.2 nv show system api versions

	nv show system api versions Show NVUE REST API versions information.
Syntax Description	N/A
Default	N/A
History	

Example	<pre>admin@nvos:~$ nv show system api versions
	         operational
	------- -----------
	nvue current
	nvue_v1 current</pre>

REST API	GET https://<ip>/nvue_v1/system/api/versions
Related Commands	nv show system api
Notes	

5.14.3.3 nv set/unset system api state

	nv set/unset system api state <enabled | disabled> Set the REST API external access state.	
Syntax Description	state	enabled, disabled
Default	enabled	
History	25.02.2002	
Example	admin@nvos:~$ nv show system api admin@nvos:~$ nv set system api state enabled	
REST API	PATCH https://<ip>/nvue_v1/system/api	
Related Commands	nv show system api	
Notes		

5.14.3.4 nv set/unset system api port

	nv set/unset system api port <port-number> Set the REST API listen port.	
Syntax Description	port-number	Port number of the remote syslog server: 1–65535
Default	443	
History	25.02.2002	
Example	admin@nvos:~$ nv set system api port 443	
REST API	PATCH https://<ip>/nvue_v1/system/api	
Related Commands	nv show system api	
Notes		

5.14.3.5 nv set system api compression

	nv set system api compression [gzip] nv unset system api compression Enable compression for NVUE REST API resonse. The unset form of the command disables compression for NVUE REST API resonse.	
Syntax Description	gzip	Enables gzip configuration in nginx for NVUED.
Default	N/A	
History	25.02.7002	
Example	admin@nvos:~$ nv set system api compression gzip	

REST API	PATCH https://<ip>/nvue_v1/system/api/?rev=<rev_id> Content-Type: application/json {"compression": "gzip"}
Related Commands	
Notes	

5.14.3.6 nv show system api mtls

	nv show system api mtls Show configured CA certificate for API mTLS connections.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	
REST API	GET https://<ip>/nvue_v1/system/api/mtls
Related Commands	
Notes	nv show system api

5.14.3.7 nv set system api mtls ca-certificate

	nv set system api mtls ca-certificate <cacert-id> Set CA certificate for API mTLS connection.	
Syntax Description	cacert-id	CA certificate ID string
Default	none	
History	25.02.2002	
Example		
REST API	PATCH https://<ip>/nvue_v1/system/api/mtls?rev= Content-Type: application/json {"ca-certificate": "cacert_id"}	
Related Commands		
Notes		

5.14.3.8 nv set system api certificate

	nv set system api certificate <cert-id> Set certificate for API connection.	
Syntax Description	cert-id	Certificate ID string
Default	none	

History	25.02.2002
Example	`admin@nvos:~$ nv set system api certificate cert cert_id`
REST API	PATCH https://<ip>/nvue_v1/system/api/mtls?rev= Content-Type: application/json {"certificate": "cert_id"}
Related Commands	
Notes	

5.15 Time Synchronization

NVOS relies on the system clock for displaying the time and timestamping messages. The date and time can be configured manually, or synchronization with Network Time Protocol (NTP) servers can be enabled.

For more information, see the following sections:

- Date and Time
- NTP

5.15.1 Date and Time

NVOS allows to control the system time zone and clock settings.

5.15.1.1 Time Zone

Default time zone is set to Coordinated Universal Time (UTC). User may change the time zone configuration by executing the following:

```
admin@nvos:~$ nv set system date-time timezone Etc/UTC
admin@nvos:~$ nv config apply
```

5.15.1.2 Clock

The system date and time can be manually changed. NTP servers configured on the switch will supersede any manually entered date-time settings.

```
admin@nvos:~$ nv action change system date-time 2024-12-24 10:25:13
```

5.15.1.3 Date and Time Commands

- Date and Time Commands

5.15.1.4 Date and Time Commands

5.15.1.4.1 nv show system date-time

	nv show system date-time Show detailed system date and time information
Syntax Description	N/A
Default	N/A
History	25.02.5002
Example	admin@nvos: ~$ nv show system date-time operational applied --- local-time Tue 2025-04-08 17:28:16 IDT universal-time Tue 2025-04-08 14:28:16 UTC rtc-time Tue 2025-04-08 14:28:17 system-clock-synchronized no ntp-service n/a rtc-in-local-tz no timezone Asia/Jerusalem Asia/Jerusalem unix-time 1744122496.6114235
REST API	GET https://<ip>/nvue_v1/system/date-time
Related commands	
Notes	

5.15.1.4.2 nv action change system date-time

	nv action change system date-time <yyyy-mm-dd> <hh:mm:ss> Sets the time and date.	
Syntax Description	hh:mm:ss	Time
	yyyy-mm-dd	Date
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action change system date-time 2021-01-01 10:10:11	
REST API	POST https://<ip>/nvue_v1/system/date-time	
Related Commands	nv show system	
Notes	Unable to change date and time in case NTP is enabled.	

5.15.1.4.3 nv set/unset system date-time timezone

	nv set system date-time timezone <timezone> nv unset system date-time timezone Sets the system time zone. The no form of the command resets time zone to its default (Etc/UTC).	
Syntax Description	timezone	A valid timezone value (e.g., Africa/Abidjan, Brazil/Acre, Africa/Accra, Chile/Continental)
Default	Etc/UTC	
History	25.02.2002 25.02.5002 Updated command syntax	
Example	`admin@nvos:~$ nv set system date-time timezone Etc/UTC`	
REST API	PATCH https://<ip>/nvue_v1/system/date-time	
Related Commands	nv show system	
Notes	The time zone may be specified in one of three ways: • A nearby city whose time zone rules to follow. The system has a large list of cities which can be displayed by the help and completion system. They are organized hierarchically because there are too many of them to display in a flat list. A given city may be required to be specified in two, three, or four words, depending on the city • An offset from Etc/GMT. This will be in the form Etc/GMT-offset, Etc/GMT+<0-14>, Etc/GMT-<1-12> • Etc/UTC (Universal Time, which is almost identical to GMT), and this is the default time zone	

5.15.2 NTP

Network Time Protocol (NTP) is a networking protocol for clock synchronization between computer systems over variable-latency data networks. NTP is intended to synchronize all participating computers to within a few milliseconds of Coordinated Universal Time (UTC) and is designed to mitigate the effects of variable network latency. NTP can usually maintain time to within tens of milliseconds over the public Internet, and can achieve better than one millisecond accuracy in local area networks under ideal conditions.

5.15.2.1 NTP Authenticate

When authentication of incoming NTP packets is enabled, the switch ensures that they come from an authenticated time source before using them for time synchronization on the switch. Authentication keys are created and marked as trusted.

To add a key to be used for authentication, take the following steps

1. Create the key.

```
admin@nvos:~$ nv set system ntp key 1
```

2. Specify the value.

```
admin@nvos:~$ nv set system ntp key 1 value mystrongpassword
```

3. Trust new added key.

```
admin@nvos:~$ nv set system ntp key 1 trusted yes
```

4. Assign the key to the server.

```
admin@nvos:~$ nv set system ntp server 10.34.1.1 key 1
```

5.15.2.2 NTP Authentication Key

An authentication key may be created and used to authenticate incoming NTP packets. For the key to be used, make sure the following is in place.

1. It should be shared with the NTP server sending the NTP packet.
2. The key should be marked as trusted.
3. NTP authentication should be enabled on the system.

5.15.2.3 NTP Commands

- NTP Commands

5.15.2.4 NTP Commands

5.15.2.4.1 nv show system ntp

	nv show system ntp Display NTP configuration. It shows the next information: status (offset, reference), authentication, listening interface, NTP state, NTP DHCP state, VRF device.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output	

Example	admin@nvos:~$ nv show system ntp operational applied reference 84.254.80.94 offset 516 ms status synchronised [listen] eth1 eth1 state enabled enabled dhcp enabled enabled vrf default default authentication disabled disabled [server] ntp0.ntp-servers.net ntp0.ntp-servers.net [server] ntp1.ntp-servers.net ntp1.ntp-servers.net [server] ntp2.ntp-servers.net ntp2.ntp-servers.net
REST API	GET https://<ip>/nvue_v1/system/ntp
Related Commands	
Notes	By default, NTP DHCP is enabled, so if the DHCP server sends out NTP servers to the clients, the switch will get it and will be synchronized with it.

5.15.2.4.2 nv show system ntp server

	nv show system ntp server [<server-id> | detail] Display NTP servers configuration and detail data. The command shows the following information: aggressive polling state, server type, authentication key, server IP address, server state, trustiness, protocol version.	
Syntax Description	server-id	Hostname or IP address of the NTP server. Show configuration for a specific configured NTP server.
	detail	Detail view: Show detail data from NTP servers.
Default	N/A	
History	25.02.2002 25.02.5002 Added the `detail` option and updated output	

Example	admin@nvos:~$ nv show system ntp server NTP server Aggressive Type KeyID Resolve as Source State Trusted Ver --- --- 37.53.71.213 server 37.53.71.213 pool 46.173.175.211 server 46.173.175.211 pool 162.159.200.123 server 162.159.200.123 pool 193.84.22.254 server 193.84.22.254 pool time.google.com disabled server 216.239.35.0 user-config enabled no 4 ua.pool.ntp.org disabled pool N/A user-config enabled no 4 admin@nvos:~$ nv show system ntp server detail NTP server Auth Delay Jitter Offset Peer State Poll Reach Ref clock Stratum Peer When --- --- --- --- --- --- --- --- --- --- --- 37.53.71.213 none 98.736 21.749 +0.337 reject-outlier 128 377 17.253.14.253 2 u 4 46.173.175.211 none 87.295 13.791 +4.291 candidate 64 377 17.253.38.125 2 u 60 162.159.200.123 none 59.550 1.401 +2.900 candidate 64 377 10.121.10.17 3 u 62 193.84.22.254 none 98.181 20.997 +11.218 reject-outlier 128 377 .PPS. 1 u 1 time.google.com none 63.111 30.026 +2.891 sys-peer 64 377 .GOOG. 1 u 65 ua.pool.ntp.org none 0.000 0.000 +0.000 reject-other 64 0 .POOL. 16 p - admin@nvos:~$ nv show system ntp server 10.7.77.134 operational applied --- iburst disabled disabled state enabled enabled trusted no no version 4 4 association-type server server resolve-as 10.7.77.134 auth-state none delay 98.736 jitter 21.749 offset +0.337 peer-state reject-outlier poll 128 reach 377 ref-clock 17.253.14.253 stratum 2 peer u when 4
REST API	GET https://<ip>/nvue_v1/system/ntp/server GET https://<ip>/nvue_v1/system/ntp/server/<server-id>
Related Commands	
Notes	

5.15.2.4.3 nv show system ntp key

	nv show system ntp key [<key-id>] Display NTP authentication keys inventory. It shows the next information: key ID, key type, obfuscated value.	
Syntax Description	key-id	NTP authentication key ID. Show the configuration of a specific authentication key.
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show system ntp key KeyID Trusted Type Value ----- ------- ---- ----- 1 yes md5 * 42 yes sha1 * admin@nvos:~$ nv show system ntp key 42 operational applied ------- ----------- ------- trusted yes yes type sha1 sha1 value * *
REST API	GET https://<ip>/nvue_v1/system/ntp/key GET https://<ip>/nvue_v1/system/ntp/key/<key-id>
Related Commands	
Notes	

5.15.2.4.4 nv set/unset system ntp

	nv set system ntp [state {enabled,disabled} | dhcp {enabled,disabled} | vrf {default} | authentication {enabled,disabled}] nv unset system ntp Update NTP global configuration. The unset form of the command returns NTP global configuration to its default.	
Syntax Description	state	NTP state configuration.
	dhcp	Use NTP servers leased from DHCP server.
	vrf	VRF to run NTP daemon in. Limited to "default" only.
	authentication	Enables NTP authentication.
Default	state	enabled
	dhcp	enabled
	vrf	default
	authentication	disabled
History	25.02.2002	
Example	admin@nvos:~$ nv unset system ntp admin@nvos:~$ nv set system ntp authentication enabled admin@nvos:~$ nv unset system ntp state	
REST API	PATCH https://<ip>/nvue_v1/system/ntp	
Related Commands	nv set system ntp server	
Notes		

5.15.2.4.5 nv set/unset system ntp listen

	nv set system ntp listen <interface-id> nv unset system ntp listen Update NTP listen configuration. The unset form of the command returns NTP listen configuration to its default.	
Syntax Description	interface-id	Interface ID
Default	eth0	
History	25.02.5002	
Example	`admin@nvos:~$ nv unset system ntp listen`	
REST API	PATCH https://<ip>/nvue_v1/system/ntp/listen	
Related Commands	nv set system ntp server	
Notes		

5.15.2.4.6 nv set/unset system ntp server

	nv set system ntp server [<server-id> [iburst {enabled,disabled} | state {enabled,disabled} | key <1-65535> | trusted {yes,no} | version {3,4} | association-type {server,pool}]] nv unset system ntp server Update the NTP servers configuration. The unset form of the command returns the NTP server configuration to its default.	
Syntax Description	server-id	Hostname or IP address of the NTP server.
	iburst	Aggressive polling of the server.
	state	Temporarily disable/enable this NTP server.
	key	Specify the key ID to securely communicate with the remote NTP server.
	trusted	Trust that NTP server. If authentication is configured this will additionally force all time updates to only use trusted servers.
	version	The NTP protocol version to communicate with the remote server.
	association-type	NTP server association type.
Default	iburst	on
	state	disabled
	trusted	no
	version	4
	association-type	server
History	25.02.2002 25.02.5002 Updated command output. Updated command option (iburst)	

Example	admin@nvos:~$ nv unset system ntp server admin@nvos:~$ nv unset system ntp server 10.10.10.10 admin@nvos:~$ nv set system ntp server 10.10.10.10 trusted yes admin@nvos:~$ nv unset system ntp server 10.10.10.10 key
REST API	PATCH https://<ip>/nvue_v1/system/ntp/server PATCH https://<ip>/nvue_v1/system/ntp/server/<server-id>
Related Commands	nv set system ntp key
Notes	When authentication of incoming NTP packets is enabled, the switch ensures that they come from an authenticated time source before using them for time synchronization on the switch. An authentication key may be created and used to authenticate incoming NTP packets. For the key to be used, make sure the following is in place. 1. It should be shared with the NTP server sending the NTP packet. 2. The key should be trusted. 3. NTP authentication should be enabled on the system

5.15.2.4.7 nv set/unset system ntp key

	nv {set,unset} system ntp key [<key-id> [trusted {yes,no} | type {md5,sha1} | value <value>]] Update the NTP keys configuration.	
Syntax Description	key-id	NTP authentication key ID.
	trusted	Trust that NTP authentication key.
	type	Authentication key type.
	value	Secret authentication key value.
Default	trusted	no
	type	md5
History	25.02.2002	
Example	admin@nvos:~$ nv unset system ntp key admin@nvos:~$ nv unset system ntp key 3 admin@nvos:~$ nv set system ntp key type sha1 admin@nvos:~$ nv unset system ntp key type	
REST API	PATCH https://<ip>/nvue_v1/system/ntp/key PATCH https://<ip>/nvue_v1/system/ntp/key/<key-id>	

Related Commands	nv set system ntp server
Notes	

5.16 User Interfaces

This section provides information on the interfaces available for users to manage and validate the status of their switch system.

5.16.1 Secure Shell (SSH) for Remote Access

NVOS uses the OpenSSH package to provide access to the system using the Secure Shell (SSH) protocol.

5.16.1.1 Overview

You can configure SSH to provide login access to the root user and to specific user accounts, limit SSH to listen on a specific VRF, and configure timeouts and session options.

- SSH server configuration changes take effect only in new SSH sessions and do not impact existing ones.
- SSH Strict Mode: By default, NVOS disables the following SSH server configurations: X11, TCP forwarding, and compression and enforces secure ciphers.

5.16.1.2 Configure Timeouts and Sessions

You can configure the following SSH timeout and session options:

- The number of login attempts allowed before rejecting the SSH session. You can specify a value between 3 and 100. The default value is 3 login attempts.
- The number of seconds allowed before login times out. You can specify a value between 1 and 600. The default value is 120 seconds.

- The TCP port numbers that listen for incoming SSH sessions. You can specify a value between 1 and 65535.
- The number of minutes a session can be inactive before the SSH server terminates the connection. The default value is 20 minutes.
- The maximum number of SSH sessions allowed per TCP connection. You can specify a value between 1 and 100. The default value is 10.

The following example configures the number of login attempts allowed before rejecting the SSH session to 10 and the number of seconds allowed before login times out to 200:

```
admin@nvos:~$ nv set system ssh-server authentication-retries 10
admin@nvos:~$ nv set system ssh-server login-timeout 200
admin@nvos:~$ nv config apply
```

The following example configures the TCP port that listens for incoming SSH sessions to 443:

```
admin@nvos:~$ nv set system ssh-server port 443
admin@nvos:~$ nv config apply
```

The following example configures the amount of time a session can be inactive before the SSH server terminates the connection to 5 minutes (300 seconds) and the maximum number of cuncurrent SSH user sessions to 5. The default **inactive-timeout** is 20 minutes and the default **max-sessions** is 100:

```
admin@nvos:~$ nv set system ssh-server inactive-timeout 5
admin@nvos:~$ nv set system ssh-server max-sessions 5
admin@nvos:~$ nv config apply
```

5.16.1.2.1 Message of the Day

When you log into the switch, NVOS shows system health information and login notifications.

Example:

```
Last login: Thu Jun 19 04:52:31 UTC 2025 from 10.20.30.40 on pts/0
Number of total successful connections since last 1 days: 6
```

SSH Login Notifications

NVOS shows the following SSH login information on the console after authentication:

- The date and time of the last successful login.
- The number of unsuccessful logins after the last successful login.
- The date and time of the last unsuccessful login.
- Changes to a user account after the last login (password, role, group, and so on).
- The location (terminal or IP) of the last successful or unsuccessful login.
- The total number of successful logins after a specific date and time.

NVOS displays login notifications for both SSH and serial connections. The information can help to detect unwanted or malicious activities, such as suspicious logins or password and role changes.

To configure the time period in days during which to show login notifications, run the `nv set system ssh-server login-record-period <days>` command. You can specify a value between 1 and 30. The default value is 1.

The following example sets the SSH login notification period to 20 days:

```
admin@nvos:~$ nv set system ssh-server login-record-period 20
admin@nvos:~$ nv config apply
```

To set the SSH login notification period back to the default value (1 day), run the `nv unset system ssh-server login-record-period` command.

To show the configured SSH login notification period, run the `nv show system ssh-server` command. See Troubleshooting below.

5.16.1.2.2 Generate and Install an SSH Key Pair

This section describes how to generate an SSH key pair on one system and install the key as an authorized key on another system.

5.16.1.2.2.1 Generate an SSH Key Pair

To generate an SSH key pair, run the `ssh-keygen` command and follow the prompts.

NVOS does not support sha1 ssh key exchange methods.

To configure the system without a password, do not enter a passphrase when prompted in the following step.

```
 admin@host01:~$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/admin/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/admin/.ssh/id_rsa.
Your public key has been saved in /home/admin/.ssh/id_rsa.pub.
The key fingerprint is:
5a:b4:16:a0:f9:14:6b:51:f6:f6:c0:76:1a:35:2b:bb cumulus@leaf04
The key's randomart image is:
+---[RSA 2048]----+
|      +.o    o   |
|     o * o . o   |
|    o + o O o    |
|     + . = O     |
|      . S o .    |
|       +    .    |
|      .   E      |
|                 |
|                 |
+-----------------+
```

5.16.1.2.2.2 Install an Authorized SSH Key

To install an authorized SSH key, you take the contents of an SSH public key and add it to the SSH authorized key file (`~/.ssh/authorized_keys`) of the user.

A public key is a text file with three space separated fields:

`<type> <key string> <comment>`

Field	Description
<type>	The algorithm you want to use to hash the key. The algorithm can be ecdsa-sha2-nistp256, ecdsa-sha2-nistp384, ecdsa-sha2-nistp521, ssh-dss, ssh-ed25519, or ssh-rsa (the default value).
<key string>	A base64 format string for the key.
<comment>	A single word string. By default, this is the name of the system that generated the key. NVUE uses the <comment> field as the key name.

The procedure to install an authorized SSH key is different based on whether the user is an NVUE managed user or a non-NVUE managed user.

NVUE Managed User

The following example adds an authorized key named prod_key to the user admin2. The content of the public key file is ssh-rsa 1234 prod_key.

```
admin@nvos:~$ nv set system aaa user admin2 ssh authorized-key prod_key key XABDB3NzaC1yc2EAAAADAQABAAABgQCvjs/
RFPhxLQMkckONg+1RE1PTIO2JQhzFN9TRg7ox7o0tfZ+IzSB99lr2dmmVe8FRWgxVjc...
admin@nvos:~$ nv set system aaa user admin2 ssh authorized-key prod_key type ssh-rsa
admin@nvos:~$ nv config apply
```

5.16.1.2.2.3 PKA-Only

This configuration allows blocking password authentication from users that have a configured authorized key.

To enable this flag, run the following:

```
admin@nvos:~$ nv set system ssh-server pka-only enabled
admin@nvos:~$ nv config apply
```

5.16.1.2.3 Troubleshooting

To show all the current SSH server configuration settings, run the NVUE nv show system ssh-server command:

```
admin@nvos:~$ nv show system ssh
                      operational  applied
--------------------  -----------  --------
authentication-retries  6            6
login-timeout           120          120
inactive-timeout        20           20
login-record-period     1            1
max-sessions            100          100
pka-only                disabled     disabled
[port]                  22           22
```

To show the TCP port numbers that listen for incoming SSH sessions, run the nv show system ssh-server port command.

5.16.1.3 SSH Commands

- SSH Commands

5.16.1.4 SSH Commands

5.16.1.4.1 nv show system ssh-server

	nv show system ssh-server Show SSH server configurations.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system ssh-server operational applied ---------- ----------- -------- authentication-retries 6 6 login-timeout 120 120 inactive-timeout 15 15 login-record-period 1 1 max-sessions 100 100 pka-only disabled disabled [port] 22 22
REST API	GET https://<ip>/nvue_v1/system/ssh-server
Related Commands	nv show system ssh-server
Notes	Applied for new connections only.

5.16.1.4.2 nv show system ssh-server port

	nv show system ssh-server port Show TCP port numbers for to listen for incoming SSH sessions.
Syntax Description	N/A
Default	N/A
History	25.02.3000

Example	admin@nvos:~$ nv show system ssh-server port Ports ----- 22
REST API	GET https://<ip>/nvue_v1/system/ssh-server/port
Related Commands	nv show system ssh-server
Notes	Applied for new connections only.

5.16.1.4.3 nv set/unset system ssh-server inactive-timeout

	nv set/unset system ssh-server inactive-timeout [<integer time>] Configure inactive timeout for ssh connections in minutes <0-35000>.	
Syntax Description	integer time	Number of minutes: 0–35000 minutes
Default	15 minutes	
History	25.02.2002 25.02.5002 Updated command syntax	
Example	admin@nvos:~$ nv set system ssh-server inactive-timeout 300	
REST API	PATCH https://<ip>/nvue_v1/system/ssh-server/inactive-timeout	
Related Commands	nv show system ssh-server	
Notes	Applied for new connections only.	

5.16.1.4.4 nv set/unset system ssh-server max-sessions

	nv set/unset system ssh-server max-sessions [<integer sessions>] Configuring the maximum number of ssh connections <3–100>.	
Syntax Description	integer sessions	Number of sessions: 3–100 sessions
Default	100	
History	25.02.2002	
Example	admin@nvos:~$ nv set system ssh-server max-sessions 10	
REST API	PATCH https://<ip>/nvue_v1/system/ssh-server/max-sessions	
Related Commands	nv show system ssh-server	
Notes	Applied for new connections only.	

5.16.1.4.5 nv set/unset system ssh-server port

	nv set/unset system ssh-server port Configure the ports for the systems ssh-server.

Syntax Description	N/A	
Default	22	
History	25.02.2002 25.02.5002 Updated command syntax	
Example	`admin@nvos:~$ nv set system ssh-server port`	
REST API	PATCH https://<ip>/nvue_v1/system/ssh-server/port	
Related Commands	nv show system ssh-server	
Notes	Multiple ports can be configured. By default, port 22 is used. Any user configuration will remove the default, the user need to configure port 22 explicitly.	

5.16.2 User Interface Commands

5.16.2.1 System Message

5.16.2.1.1 nv show system message

	nv show system message Display banner messages for different terminal session events: pre-login, post-login and post-logout messages.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output	

Example	`admin@nvos:~ nv show system message`
	`operational applied`
	`pre-login pre_login_message pre_login_message`
	`post-login post_login_message post_login_message`
	`post-logout post_logout_message post_logout_message`	

REST API	GET https://<ip>/nvue_v1/system/message	

Related Commands	nv set system message pre-login nv set system message post-login
Notes	Printed as raw, unformatted output

5.16.2.1.2 nv set/unset system message pre-login message

	nv set system message pre-login {message} nv unset system message pre-login Set/unset the pre-login message.	
Syntax Description	message	The new pre-login message to set
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system message pre-login new-pre admin@nvos:~$ nv unset system message pre-login	
REST API	PATCH https://<ip>/nvue_v1/system/message	
Related Commands	nv show system message	
Notes		

5.16.2.1.3 nv set/unset system message post-login message

	nv set system message post-login {message} nv unset system message post-login Set the post-login message. The unset form of the command unsets the post-login message.	
Syntax Description	message	The new post-login message to set
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system message post-login new-post admin@nvos:~$ nv unset system message post-login	
REST API	PATCH https://<ip>/nvue_v1/system/message	
Related Commands	nv show system message	
Notes		

5.16.2.1.4 nv set/unset system message post-logout

	nv set system message post-logout <message> nv unset system message post-logout Set/unset the post-logout message.	
Syntax Description	message	The new post-logout message to set

Default	N/A
History	25.02.2002
Example	`admin@nvos:~$ nv set system message post-logout Goodbye`
REST API	PATCH https://<ip>/nvue_v1/system/message
Related Commands	nv show system message
Notes	

5.16.2.2 Serial-Console

5.16.2.2.1 nv show system serial-console

	nv show system serial-console Show system serial console.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output 25.02.5002 Updated command output

Example	`admin@nvos:~$ nv show system serial-console`
	`operational applied`
	`------------------ ----------- --------`
	`sysrq-capabilities disabled disabled`

REST API	GET https://<ip>/nvue_v1/system/serial-console
Related Commands	nv set system serial-console sysrq-capabilities
Notes	Applied after reconnection only.

5.16.2.2.2 nv set/unset system serial-console sysrq-capabilities

	nv set/unset system serial-console sysrq-capabilities Enables or disables SysRq key capabilities.
Default	Disabled
History	25.02.2002
Example	`admin@nvos:~$ nv set system serial-console sysrq-capabilities enabled`
REST API	PATCH https://<ip>/nvue_v1/system/ssh-server
Related Commands	nv show system serial-console
Notes	

5.16.2.3 System CLI

5.16.2.3.1 nv show system cli

	nv show system cli Show System CLI.
Syntax Description	N/A
Default	N/A
History	25.02.5002
Example	admin@nvos:~$ nv show system cli operational applied inactive-timeout 20 20
REST API	GET https://<ip>/nvue_v1/system/cli
Related Commands	nv set system cli inactive-timeout
Notes	

5.16.2.3.2 nv set/unset system cli inactive-timeout

	nv set system cli inactive-timeout [minutes] nv unset system cli inactive-timeout Configure inactive timeout for CLI.	
Syntax Description	minutes	The number of minutes before CLI becomes inactive. Maximum timeout: 86400 (minutes)
Default	20	
History	25.02.5002	
Example	admin@nvos:~$ nv set system cli inactive-timeout 300	
REST API	PATCH https://<ip>/nvue_v1/system/cli	
Related Commands		
Notes		

5.17 Zero-Touch Provisioning

Zero-touch Provisioning (ZTP) automates the initial configuration of switch systems at boot time. It helps minimize manual operation and reduce customer initial deployment costs. ZTP allows for automatic upgrade of the switch with a specified OS image, setting up a switch configuration, and checking connectivity to external resources.

A switch configuration is applied by using either of the following two formats: YAML format file configuration or CLI commands in the regular text file.

The user can create a YAML configuration file by saving the running configuration by “nv config save” and later uploading that file to an external provisioning server.

The textual configuration file can be created by running “nv config show -o commands” and pasting the output to the text file. That text file can be edited later to add more commands or replace the ones in the list.

ZTP can execute custom provisioning scripts. These scripts can be of any type, must include a shebang, and will run during the ZTP process.

5.17.1 Running DHCP-ZTP

There is no explicit command to enable ZTP. It is enabled by default. Disabling it is performed by a user-initiated configuration save (using the command “nv config save”).
there are two ways to re-enable ZTP:

1. Run the `reset factory` command: This clears the entire switch configuration and reboots the system. This is the recommended method.
2. Run `nv action run system ztp` : This command will initiate the ZTP process only if there is no existing configuration on the system. Unlike the factory reset, this command does not automatically remove the startup configuration; the user must ensure the configuration is cleared before running it.

ZTP is based on DHCP. For ZTP to work, the software enables DHCP by default on all its management interfaces. The switch OS requests option 66 (tftp-server-name) and/or 67 (bootfile-name) from the DHCPv4 server or option 59 (bootfile-url) from the DHCPv6 server and

waits for the DHCP responses containing the ZTP JSON configuration file URL. See (25.03.0600-VR-0.6) Zero-Touch Provisioning#ZTP Configuration File section for more information.

The DHCP server must be configured to send back the URL to the ZTP configuration file. For DHCPv4 there are two options supported: Option 66 and Option 67. Providing both options by DHCP server will result in combining them to build TFTP ULR to ZTP configuration file. When using Option 66 the TFTP protocol prefix is omitted. Providing only Option 67 will be considered as URL to the location of the ZTP configuration file.

For DHCPv6, the only option supported is Option 67. It will contain the complete URL to the ZTP JSON configuration file. The format of all the options is a string. Below is a summary table:

DHCP Option	Name	Description
66	tftp-server-name	TFTP-Server address. If specified by server, must be used together with option 67.
67	bootfile-name	URL to download the ZTP JSON configuration file. It can also specify the ZTP JSON file path on TFTP server.
59	dhcp6.bootfile-url	URL to download the ZTP JSON configuration file.

Examples of Options for DHCPv4:

```
option tftp-server-name "198.51.100.8";
option bootfile-name "ztp.json";
```

⚠ It will result in `tftp://192.51.100.1/ztp.json`

Example of only Option 67 for DHCPv4:

```
option bootfile-name "scp://<user>:<pass>@198.51.100.8/ztp.json";
```

Example of only Option 67 for DHCPv6:

```
option dhcp6.bootfile-url "http://[2001:db8::8]/ztp.json";
```

5.17.2 Dynamic Content Configuration

When a switch requests a file specified by a URL, it is expected that the server which is handling the request knows which file to be returned. For example, when a switch requests for configuration file, the server needs to be able to give out a configuration file for the requesting switch. This selection of the file can be done in two ways:

- Provisioning Server side: DHCP Options, HTTP Headers
- ZTP Client side: Dynamic URL (see ZTP Configuration File Section)

5.17.2.1 DHCP Options

To have the DHCP server discern the proper files based on switch-specific information, the NOS must provide identifying information for the server to classify the switches. NOS provides the following identifying information:

- Product Name—Option 61
- Serial Number—Option 61
- Vendor Class Identifier—Option 77 (Option 15 for DHCPv6)

Upon receiving such DHCP requests from a client, the server should be able to map the switch-specific information to the target file URLs according to predefined rules.

The following are the DHCP options sent by the NVOS switch in the DHCP request:

DHCP Option	Name	Description
61	dhcp-client-identifier	Used to uniquely identify the switch initiating DHCP request. NVOS switches set this value: For eth0: "NVOS##*product-name*##*serial-no*" For eth1: "NVOS##*product-name*##*serial-no*##*interface-name*"
77	user-class	Used to optionally identify the type or category of user or applications it represents. NVOS switches set this value to "NVOS-ZTP".
15	dhcp6.user-class	Used to optionally identify the type or category of user or applications it represents. NVOS switches set this value to "NVOS-ZTP".

5.17.2.2 HTTP Headers

For a server to make the decision on which file to serve out, it requires uniquely identifiable information about the requesting switch. Then, appropriate logic can be implemented on the server side to process the provided information, and identify and give out the requested file.

All HTTP/HTTPS requests made during ZTP contain switch identification information as part of HTTP headers. Below is the information that is included.

Header	Value	Example
User-Agent	NVOS-ZTP/0.1	
PRODUCT-NAME	*String specifying the switch model*	
SERIAL-NUMBER	*String specifying the manufacture provided serial number*	MT1234X56789
BASE-MAC-ADDRESS	*Ethernet MAC Address assigned to the switch by the manufacturer*	12:34:56:AB:CD:EF
NVOS-VERSION	*Version string as seen in 'nv show system version' command*	nvos-25.02.xxxx
NVUE-VERSION	*Version string as seen in 'nv show system version packages installed python3-nvue' command*	2.X.X.X-nvos

5.17.3 ZTP Configuration

5.17.3.1 ZTP Configuration File

For ZTP to automate the provisioning process a user needs to provide a JSON configuration file. The ZTP JSON configuration file consists of configuration sections (objects). Each section has its own options. In other words, ZTP JSON configuration file is an instruction on how to perform provisioning and from where to get provisioning data. Here is an example of the ZTP JSON configuration file:

```
{
 "ztp": {
    "01-image": {
      "install": {
        "url": "http://198.51.100.2/images/nvos.bin"
      },
      "uninstall": true
    },
    "02-commands-list": {
      "url": "sftp://user:password@198.51.100.3/configs/commands.txt",
      "clear-config": "true"
    },
    "03-startup-file": {
      "url": "scp://user:password@198.51.100.4/configs/config.yaml",
      "clear-config": false
    },
    "04-connectivity-check": {
      "ping-hosts": [ "198.51.100.5", "localhost" ]
      "ping-count": 10,
      "ignore-result": true
    },
    "05-provisioning-script": {
      "url": "scp://user:password@198.51.100.5/scripts/script.sh",
      "timeout": 300
    }
  }
}
```

The main section is “ztp” and it must always be present in the ZTP JSON configuration file. Inside the main section may have the following configuration sections:

- image—to manage system images on the switch
- commands-list—to apply CLI commands in textual form
- startup-file—to apply YAML formatted configuration
- connectivity-check—to check a connectivity to predefined location

The configuration sections are processed by ZTP software in lexical order so to control the order of execution, a “xx-“ prefix to the section names is allowed (e.g., 02-commands-list).

Each configuration section of ZTP JSON includes some common parameters that can be used to influence its execution. The default value for a parameter is assumed when the parameter is not specified in the configuration section:

- description: Optional free-form text string used to describe a configuration section defined in the ZTP JSON configuration file.
- ignore-result (default: false):
    - true—ZTP service marks status as SUCCESS even if an error is encountered while processing this individual section.
    - false—ZTP service marks status as FAILED if an error is encountered while processing this individual section.
- halt-on-failure (default: false):

- true—If configuration section result is FAILED, ZTP service stops and exits immediately marking ZTP status as FAILED. No other configuration sections are processed. User intervention is needed to restart ZTP.
- false—ZTP service moves on to next configuration section.

And the main “ztp” section has its own parameter:

- restart-ztp-no-config (default: false):
  - true—ZTP procedure is restarted if switch startup configuration file is not present after the completion of processing the configuration sections defined in the ZTP JSON configuration file. This happens up to 10 times.
  - false—ZTP service exits after processing all of the configuration sections defined in the ZTP JSON configuration file even if the Switch startup configuration file is not present.
- restart-ztp-on-failure (default: true):
  - true—ZTP procedure is restarted if the result of ZTP is FAILED after processing all of the configuration sections defined in the ZTP JSON configuration file. This happens up to 10 times.
  - false—ZTP service exits after processing all of the configuration sections defined in the ZTP JSON configuration file.

ZTP service exits and marks the status as FAILED if any errors are encountered while parsing the ZTP JSON configuration file. It is encouraged to check for any JSON format correctness before rolling out the ZTP JSON configuration file for use. When processing a configuration section and provided data is found to be insufficient, it is marked as failed and ZTP moves on to the next section.

5.17.3.2 ZTP Configuration image

The *image* configuration section is used for image management on a switch. It can be used to install and uninstall system images.

Example config section to install a new image and boot into it:

```
"image": {
    "install": {
        "url": "http://198.51.100.2/images/nvos.bin",
        "skip-reboot ": false
    }
}
Example config section to uninstall a NVOS image from the Switch:
"image": {
    "uninstall": true
}
```

Following is the list of parameters supported in image configuration section and their brief description with a default values. The default value for a parameter is assumed when the parameter is not used:

- install—Used to install an image using URL.
  - url/dynamic-url—Specifies the URL string from where the system image file has to be downloaded in the form of url or dynamic-url object.
  - skip-reboot (default: false)—Specifies if a switch reboot operation is performed immediately after installing a new switch image. Reboot is skipped when set to true.
- uninstall (default: false)—Used to uninstall an existing image on the disk.

- true—Uninstall the image from the second partition.
- false—Do nothing.

uninstall is first processed followed by install if both are defined.

5.17.3.3 ZTP Configuration startup-file

The *startup-file* configuration section is used to apply the switch configuration in form of YAML file and apply the configuration. The YAML file format of the configuration is the same as switch startup configuration file. Here is an example of startup-file configuration section:

```
"startup-file": {
    "url": "http://198.51.100.3/startup.yaml",
    "clear-config": true,
    "save-config": true
}
```

Following is the list of parameters supported by the startup-file configuration section:

- url/dynamic-url—Define the URL string from where the startup.yaml file has to be downloaded in the form of url or dynamic-url object.
- clear-config (default: true)—Use this to specify if the existing configuration has to be cleared before loading the download startup.yaml file content . When set to true, ZTP replaces all current configuration with startup.yaml file content. When set to false it merges the configurations.
- save-config (default: false)—Use this to perform config save command after loading the downloaded startup.yaml file.

Setting save-config parameter to true will save the configuration and hence disable the ZTP at the end of the ZTP session.

5.17.3.4 ZTP Configuration commands-list

The *commands-list* configuration section is used to apply switch configuration in form of CLI text commands. The list of commands is provided in text file which should be downloaded and applied. Commands' file must contain only non-interactive commands, otherwise it will fail entire *commands-list* section. It is user responsibility to make sure commands are non-interactive, for example user should provide “force / -y / -n” or similar flags to the interactive CLI to prevent AYS (Are You Sure?) questions.

Running the `system reboot` command restarts the switch and restarts the ZTP flow for the current section from the beginning, which can result in a boot loop. Keep this in mind when defining the commands-list text file. To reboot without entering a boot loop, use the `reboot` configuration parameter, which reboots the system only after all commands have been executed.

Example of commands-list configuration section:

```
"commands-list": {
    "url": "http://198.51.100.4/commands.txt",
    "clear-config": true,
    "save-config": true
}
```

Following is the list of parameters supported by the commands-list configuration section:

- url/dynamic-url—Define the URL string from where the commands-list file has to be downloaded in the form of url object or dynamic-url object.
- clear-config (default: true)—Use this to specify if the existing configuration has to be cleared before applying file content. If true, it applies empty (default) configuration prior executing commands.
- save-config (default: false)—Use this to perform a *config save* command after applying downloaded commands list.
- reboot (default: false)—Use this to reboot the system after executing all commands.

Setting save-config parameter to true will save the configuration and hence disable the ZTP at the end of the ZTP session. The same behavior applies to “nv config save” command specified in commands-list text file.

5.17.3.5 ZTP Configuration connectivity-check

The *connectivity-check* section is used to ping a remote host and verify if the switch is able to reach the remote host. It is possible to ping multiple hosts and the plugin result is marked as failed even if ping to one of the specified host fails.

```
"connectivity-check": {
    "ping-hosts": [ "198.51.100.5", "localhost" ]
}
```

Following is the list of objects supported by the connectivity-check section:

- ping-hosts—List of IPv4 hosts to ping.
- ping6-hosts—List of IPv6 hosts to ping.
- retry-interval (default: 5)—Specify a timeout, in seconds, before retrying ping to a host.
- retry-count (default: 12)—Stop ping to a host and move on to the next host specified in the list after retrying specified count of times.
- ping-count (default: 3)—Stop after sending *count* ECHO_REQUEST packets. With *deadline* option, ping waits for *count* ECHO_REPLY packets, until the timeout expires.
- deadline (default: N/A)—Specify a timeout, in seconds, before ping exits regardless of how many packets have been sent or received. In this case ping does not stop after *count* packet are sent, it waits either for *deadline* expire or until *count* probes are answered or for some error notification from network.
- timeout (default: N/A)—Time to wait for a response, in seconds. The option affects only timeout in absence of any responses, otherwise ping waits for two RTTs (Round Trip Time).

5.17.3.6 ZTP Configuration provisioning-script

The `provisioning-script` plugin downloads and executes a file.

The downloaded file must meet the following requirements:

- Include a shebang, such as `bash` or `python`.
- Contain only non-interactive commands.
- Return an exit code of `0` upon successful execution.

A non-zero exit code causes the `provisioning-script` section to fail.

5.17.3.6.1 Configuration Example

```
// ztp.json

{
    "ztp": {
        "1-provisioning-script": {
            "url": "scp://user:password@server/path/to/script",
            "timeout": 300,
            "parameters": {
                "--nvos_image": "/images/nvos-3.10.bin",
                "--site": "dc1",
                "-n": 3,
                "--verbose": true
            }
        }
    }
}
```

5.17.3.6.2 Supported Objects

Object	Description
`url` / `dynamic-url`	Defines the URL from which to download the file.
`timeout`	Specifies the timeout, in seconds. The default value is `300`. If the timeout is exceeded, the script is terminated and the section fails. This object is optional.
`parameters`	Optional object containing name/value pairs that are passed to the script as command-line arguments. Each name represents the option flag itself, such as `-i` or `--image`, and is passed unchanged with its corresponding value. When this object is absent or empty, the script runs without arguments.
Common objects	Supports common ZTP objects, such as `halt-on-failure`.

5.17.3.6.3 Parameter Rules

The `parameters` object must comply with the following rules:

- Parameter names must be valid command-line flags:
  - A single dash followed by one letter, such as `-i`
  - Two dashes followed by a name that begins with a letter and contains letters, digits, underscores, or hyphens, such as `--image` or `--nvos_image`
- Parameter names are passed unchanged. The plugin does not add dashes.
- Parameter values must be scalar values:
  - String
  - Number
  - Boolean

- Arrays, objects, and `null` values are not supported.
- A maximum of 20 parameters is supported.

A malformed `parameters` object causes the section to fail before the file is downloaded. Examples of malformed configurations include:

- An invalid `parameters` object type
- An invalid parameter name
- A non-scalar parameter value
- More than 20 parameters

The failure includes a descriptive error message.

5.17.3.6.4 String Value Escaping

String values follow standard JSON escaping rules:

- To include a double quotation mark, use `\"`.
- To include a backslash, use `\\`.

For example:

```
"parameters": {
    "--message": "hello \"world\"",
    "--path": "C:\\nvos\\img"
}
```

5.17.4 ZTP and OS Upgrade

Software upgrade from non-ZTP versions to ZTP versions and vice versa is supported. When upgrading from a non-ZTP version, ZTP is disabled because ZTP is always assumed to start with an empty configuration.

5.17.5 Example Configurations

5.17.5.1 DHCPv4 Configuration

The following is a configuration example for ISC DHCPv4 server:

```
host master {
    hardware ethernet 12:34:56:AB:CD:EF;
    fixed-address 192.51.100.201;
    option bootfile-name "scp://<user>:<password>@192.51.100.8/ztp.json";
}
```

5.17.5.2 DHCPv6 Configuration

The following is a DHCPv6 configuration example:

```
host master {
    ........
    option dhcp6.bootfile-url "http://[2001:db8::8]/ztp.json";
}
```

5.17.5.3 ZTP Configuration File Example

```
{
  "ztp": {
    "01-image": {
      "uninstall": true
    },
    "02-commands-list": {
      "url": "sftp://user:password@198.51.100.3/configs/commands.txt",
      "clear-config": "true"
    },
    "03-startup-file": {
      "url": "scp://user:password@198.51.100.4/configs/config.yaml"
    },
    "04-connectivity-check": {
      "ping-hosts": [ "198.51.100.5", "localhost" ],
    }
  }
}
```

5.17.5.4 Commands List Example

```
# Disable password hardening
nv set system security password-hardening state disabled

# Configure log rotation rules
nv set system log rotation frequency weekly

# Configure system message
nv set system message pre-login Hello

# Applying config
nv config apply -y
```

5.17.5.5 YAML Formatted File Configuration

```
- set:
    system:
      log:
        rotation:
          size: 30.0
      ntp:
        dhcp: disabled
        server:
          localntpserver: {}
          ua.pool.ntp.org:
            association-type: pool
      security:
        password-hardening:
          state: disabled
```

5.17.5.6 Provisioning Script Example

```
#!/bin/python
print("Hello world from Provisioning Script!")
```

5.17.6 Zero-Touch Provisioning Commands

- Zero-Touch Provisioning Commands

5.17.7 Zero-Touch Provisioning Commands

- 5.17.7.1 nv show system ztp
- 5.17.7.2 nv set system ztp config-save
- 5.17.7.3 nv action system ztp

5.17.7.1 nv show system ztp

	nv show system ztp Shows global ZTP status and the status for each section.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system ztp operational applied runtime 03m 53s service enabled source dhcp6-opt59 status success config-save disabled disabled ZTP stages Stage Exit Status Ignore Result Runtime Status 00-image success no 31s success 01-connectivity-check success no 03s success 03-startup-file success no 51s success 05-commands-list success no 31s success 10-connectivity-check success no 03s success 11-provisioning-script success no 02s success admin@nvos:~$ nv show system ztp operational applied service inactive state disabled status not-started config-save disabled disabled ZTP stages No Data
REST API	GET https://<ip>/nvue_v1/system/ztp
Related Commands	nv set system ztp config-save
Notes	

5.17.7.2 nv set system ztp config-save

	nv set system ztp config-save <enabled | disabled> Sets the ZTP configuration.	
Syntax Description	enabled	Enables “nv config save”.
	disabled	Disables “nv config save”.
Default	Disabled	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv set system ztp config-save enabled
	admin@nvos:~$ nv config apply
	admin@nvos:~$ nv config save</pre>	

REST API	PATCH https://<ip>/nvue_v1/system/ztp	
Related Commands	nv show system ztp	
Notes	When ZTP is active, `nv config save` is suppressed because it may interfere with ZTP operation. Therefore, after running `nv set system ztp config-save enabled` and `nv config apply`, running `nv config save` disables ZTP as a result of the database save. There is one exception: if the ZTP JSON configuration file specifies `"save-config": true`, the configuration is saved automatically after ZTP finishes, and ZTP is then disabled. In this case, `nv set system ztp config-save` is not required.	

5.17.7.3 nv action system ztp

	nv action <abort | run> system ztp [force] ZTP action commands to interact with ZTP service.	
Syntax Description	abort	Abort ZTP session. This command is used to interrupt ongoing ZTP session.
	run	Rerun ZTP flow. Use this command to manually restart a new ZTP session from scratch or from when it failed or aborted.
	force	This option is used to skip AYS dialog. By default, each action command asks the user to confirm the execution of the specified command.
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv action abort system ztp
	The operation will perform abort of the ZTP.
	Type [y] to perform abort of the ZTP.
	Type [N] to cancel an action.
	
	Do you want to continue? [y/N] y
	Action executing ...
	Aborting ZTP session
	Action executing ...
	Action succeeded
	
	admin@nvos:~$ nv action run system ztp
	The operation will perform rerun of the ZTP.
	Type [y] to perform rerun of the ZTP.
	Type [N] to cancel an action.
	
	Do you want to continue? [y/N] y
	Action executing ...
	Rerunning ZTP session
	Action executing ...
	Action succeeded</pre>

REST API	POST https://<ip>/nvue_v1/system/ztp
Related Commands	nv show system ztp
Notes	• When ZTP is active, `nv config save` is suppressed because it may interfere with ZTP operation. Therefore, after running `nv set system ztp config-save enabled` and `nv config apply`, running `nv config save` disables ZTP as a consequence of the database save. Exception: If the ZTP JSON configuration file specifies `'save-config': true`, the configuration is automatically saved after ZTP completes successfully, and ZTP is then disabled. In this case, `nv set system ztp config-save enabled` is not required.

6 Chassis Management

The chassis manager provides the user access to the following information:

Accessible Parameters	Description
switch temperatures	Displays system’s temperature
switch leakage	Displays leakage sensors' status
fan unit	Displays system fans’ status
power unit	Displays system power consumers
Flash memory	Displays information about system memory utilization.

Additionally, it monitors:

- AC power to the PSUs
- DC power out from the PSUs
- Chassis failures
- Leakage detection from the switch tray

6.1 System Health Monitor

The system health monitor scans the system to decide whether or not the system is healthy. When the monitor discovers that one of the system's modules (leaf, spine, fan, or power supply) is in an unhealthy state or has returned from an unhealthy state, it notifies the users through the following methods:

- System logs—accessible to the user at any time as they are saved permanently on the system
- Status LEDs—changed by the system health monitor when an error is found in the system and is resolved

6.2 Leakage Sensors

The system will have 6 leakage sensors located in different locations. Once sensors detect leakage, NVOS will publish an event immediately and update system health accordingly. User will be able to see on which sensor it was detected using the 'nv show platform environment leakage' show command. Once leakage is redeemed, the sensors can re-arm automatically without the need to clear the sensors' leak state and it takes up to 1 hour.

6.3 Chassis Management Commands

- Chassis Information and Inventory
- Environment
- Software
- Transceiver

6.4 Chassis Information and Inventory

NVOS provides commands to display general platform information and inventory details.

6.4.1 Chassis Information and Inventory Commands

- Chassis Information and Inventory Commands

6.4.2 Chassis Information and Inventory Commands

6.4.2.1 nv show platform

	nv show platform Displays the types of data available under more specific platform commands.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.5002 Updated the output field "product-name" with "system type"
Example	admin@nvos:~$ nv show platform operational --- system-mac B0:XX:XX:XX:XX:00 manufacturer Nvidia cpu x86_64 Intel(R) Core(TM) i3-8100H CPU @ 3.00GHz x4 memory 15.0 GB disk-size 149.1 GB port-layout 73 x 800G part-number 920-9B31-RX-5M0-NS serial-number MT2419X0XX13 asic-model Quantum3 asic-revision 0x00A0 system-uuid 9XXXX582-XXXX-11ef-XXXX-b0cf0e6d9000 system-type QM3400
REST API	GET https://<ip>/nvue_v1/platform
Related Commands	nv show platform environment nv show platform firmware nv show platform software
Notes	

6.4.2.2 nv show platform chassis-location

	nv show platform chassis-location Display chassis location information.
Syntax Description	N/A

Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show platform chassis-location operational --- tray-index 0 slot-number 5 chassis-sn 999WWYY123456 topology-id GB200 NVL36
Related Commands	
Notes	

6.4.2.3 nv show platform inventory

	nv show platform inventory {<inventory-id>} Display the status of all platform components. Includes the following fields: hw-version, model, serial, state and type.	
Syntax Description	inventory-id	Display the status of a single platform component (with the same fields as the general command).
Default	N/A	
History	25.02.2002 25.02.3000: Updated command output	
Example	admin@nvos:~$ nv show platform inventory HW Version Model Serial State Type BMC N/A P3809 1333624062905 ok bmc FAN1/1 N/A N/A N/A ok fan FAN1/2 N/A N/A N/A ok fan FAN2/1 N/A N/A N/A ok fan FAN2/2 N/A N/A N/A ok fan FAN3/1 N/A N/A N/A ok fan FAN3/2 N/A N/A N/A ok fan FAN4/1 N/A N/A N/A ok fan FAN4/2 N/A N/A N/A ok fan SWITCH A1 692-9K36F-A5MV-JQS MT2441XXXX90 ok switch admin@nvos:~$ nv show platform inventory SWITCH operational --- state ok hardware-version A2 model 692-9K36F-00MV-JS0 serial MT2416X02630 type switch	
REST API	GET https://<ip>/nvue_v1/platform/inventroy/ GET https://<ip>/nvue_v1/platform/inventroy/{inventroy-id}	
Related Commands	nv show platform	
Notes		

6.5 Environment

NVOS includes robust features for monitoring and managing the platform's environmental conditions, ensuring optimal performance and system reliability. These features provide detailed insights into critical hardware components and allow users to take control of specific environmental aspects.

6.5.1 Key Functionalities

- Fan Monitoring: Retrieve real-time status and performance of system fans.
- Power Supply Management: Monitor the health and functionality of power supply units (PSUs).
- Temperature Monitoring: Check temperature readings from various sensors.
- Voltage Monitoring: Inspect voltage levels and sensor data.
- Leakage Detection: Identify potential leakage scenarios.
- LED Control: Manage and monitor platform LEDs.

6.5.2 Environment Commands

- Environment Commands

6.5.3 Environment Commands

6.5.3.1 nv show platform environment

	nv show platform environment Displays the types of data available under more specific platform environment commands.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show platform environment Name Type State ------------------------------ ---------- ----- ASIC temperature ok Ambient-Fan-Side-Temp temperature ok CPU-Pack-Temp temperature ok FAN1 led off FAN1/1 fan ok FAN1/2 fan ok FAN2 led off FAN2/1 fan ok FAN2/2 fan ok FAN3 led off FAN3/1 fan ok FAN3/2 fan ok FAN4 led off FAN4/1 fan ok FAN4/2 fan ok FAN5 led off FAN5/1 fan ok FAN5/2 fan ok FAN6 led off FAN6/1 fan ok FAN6/2 fan ok HSC-VinDC-In voltage ok HSC-VinDC-Out voltage ok PDB-1-Conv-In-1 voltage ok PDB-1-Conv-Out-1 voltage ok PDB-2-Conv-In-1 voltage ok PDB-2-Conv-Out-1 voltage ok PDB-3-Conv-In-1 voltage ok PDB-3-Conv-Out-1 voltage ok PDB-4-Conv-In-1 voltage ok PDB-4-Conv-Out-1 voltage ok PMIC-1-12V-VDD-ASIC1-In-1 voltage ok PMIC-1-ASIC1-VDD-Out-1 voltage ok PMIC-2-12V-HVDD-DVDD-ASIC1-In-1 voltage ok PMIC-2-ASIC1-DVDD-PL0-Out-2 voltage ok PMIC-2-ASIC1-HVDD-PL0-Out-1 voltage ok PMIC-3-12V-HVDD-DVDD-ASIC1-In-1 voltage ok PMIC-3-ASIC1-DVDD-PL1-Out-2 voltage ok PMIC-3-ASIC1-HVDD-PL1-Out-1 voltage ok PMIC-4-12V-VDD-ASIC2-In-1 voltage ok PMIC-4-ASIC2-VDD-Out-1 voltage ok PMIC-5-12V-HVDD-DVDD-ASIC2-In-1 voltage ok PMIC-5-ASIC2-DVDD-PL0-Out-2 voltage ok PMIC-5-ASIC2-HVDD-PL0-Out-1 voltage ok PMIC-6-12V-HVDD-DVDD-ASIC2-In-1 voltage ok PMIC-6-ASIC2-DVDD-PL1-Out-2 voltage ok PMIC-6-ASIC2-HVDD-PL1-Out-1 voltage ok PMIC-7-12V-MAIN-In-1 voltage ok PMIC-7-CEX-VDD-Out-1 voltage ok PMIC-8-COMEX-VDD-MEM-In-1 voltage ok PMIC-8-COMEX-VDD-MEM-Out-1 voltage ok SODIMM-1-Temp temperature ok STATUS led off UID led off
REST API	GET https://<ip>/nvue_v1/platform/environment
Related Commands	nv show platform environment fan nv show platform environment led nv show platform environment temperature
Notes	

6.5.3.2 nv show platform environment fan

	nv show platform environment fan <fan-id> Displays the maximum, minimum, current speed and state for one or all fans in the system.	
Syntax Description	fan-id	Name of the fan whose status to display. If not entered, all fans are displayed.
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show platform environment fan Name Fan State Current Speed (RPM) Max Speed Min Speed Fan Direction ------ --------- ------------------- --------- --------- ------------- FAN1/1 ok 23045 33000 6000 B2F FAN1/2 ok 21634 33000 6000 B2F FAN2/1 ok 23557 33000 6000 B2F FAN2/2 ok 22085 33000 6000 B2F FAN3/1 ok 22555 33000 6000 B2F FAN3/2 ok 22085 33000 6000 B2F FAN4/1 ok 22555 33000 6000 B2F FAN4/2 ok 22085 33000 6000 B2F FAN5/1 ok 22555 33000 6000 B2F FAN5/2 ok 22085 33000 6000 B2F FAN6/1 ok 22555 33000 6000 B2F FAN6/2 ok 21634 33000 6000 B2F admin@nvos:~$ nv show platform environment fan FAN1/1 operational ------------ ----------- state ok current-speed 22555 min-speed 6000 max-speed 33000 direction B2F
REST API	GET https://<ip>/nvue_v1/platform/environment/fan GET https://<ip>/nvue_v1/platform/environment/fan/{fan-id}
Related Commands	nv show platform environment
Notes	

6.5.3.3 nv show platform environment led

	nv show platform environment led <led-id> Displays the status of one or all LEDs in the system.	
Syntax Description	led-id	Name of the LED whose status to display. If not entered, all LEDs are displayed.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show platform environment led LED Name LED Color -------- --------- FAN1 off FAN2 off FAN3 off FAN4 off FAN5 off FAN6 off STATUS green UID blue admin@nvos:~$ nv show platform environment led UID operational ----- ----------- color blue	
REST API	GET https://<ip>/nvue_v1/platform/environment/led GET https://<ip>/nvue_v1/platform/environment/led/{led-id}	
Related Commands	nv show platform environment	
Notes		

6.5.3.4 nv show platform environment psu

	nv show platform environment psu <psu-id> Displays the capacity, current, power, voltage, and state for one or all PSUs in the system.	
Syntax Description	psu-id	Name of the PSU whose status is to be displayed. If psu-is is not specified, all PSUs are displayed.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show platform environment psu Name Capacity [W] Current [A] Power [W] Voltage [V] State ---- ------------ ------------ --------- ------------ ------ PSU1 N/A N/A N/A N/A absent PSU2 2200.00 0.53 120.84 228.00 ok PSU3 2200.00 0.58 132.67 228.75 ok PSU4 N/A N/A N/A N/A absent PSU5 N/A N/A N/A N/A absent PSU6 2200.00 0.64 145.60 227.50 ok PSU7 2200.00 0.52 118.95 228.75 ok PSU8 N/A N/A N/A N/A absent admin @nvos :~$ nv show platform environment psu PSU3 operational -------- ----------- state ok capacity 2200.00 current 0.60 voltage 228.75 power 137.25	
REST API	GET https://<ip>/nvue_v1/platform/environment/psu GET https://<ip>/nvue_v1/platform/environment/psu/<psu-id>	
Related Commands	nv show platform environment nv show platform ps-redundancy	
Notes	• Depending on the system's power supply redundancy policy, not all PSUs must be present. Therefore, absent PSUs may not cause a system health issue. • All PSU sockets must be populated either by a PSU or by a dummy PSU. • PSUs without power will be in 'absent' state.	

6.5.3.5 nv show platform environment temperature

	nv show platform environment temperature <sensor-id> Shows platform temperature and displays temperature information from different platform sensors.	
Syntax Description	sensor-id	Name of the sensor whose data to display. If not entered, all sensors are displayed.
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show platform environment temperature Name Cur Temp (°C) Crit Temp Max Temp Min Temp State ---------- ---------- ---------- -------- -------- ----- ASIC1 44.00 120.00 105.00 ok ASIC2 44.00 120.00 105.00 ok ASIC3 45.00 120.00 105.00 ok ASIC4 45.00 120.00 105.00 ok CPU-Pack-Temp 30.75 100.00 95.00 ok Drive-Temp 25.85 94.85 89.85 ok HSC-VinDC-Temp 32.44 255.94 255.94 ok PDB-Conv-1-Temp 38.00 125.00 ok PDB-Conv-2-Temp 36.00 125.00 ok PMIC-1-Temp 39.00 125.00 105.00 ok PMIC-2-Temp 40.00 125.00 105.00 ok PMIC-3-Temp 43.00 125.00 105.00 ok PMIC-4-Temp 45.00 125.00 105.00 ok PMIC-5-Temp 45.00 125.00 105.00 ok PMIC-6-Temp 43.00 125.00 105.00 ok PMIC-7-Temp 42.00 125.00 105.00 ok PMIC-8-Temp 44.00 125.00 105.00 ok PMIC-9-Temp 43.00 125.00 105.00 ok PMIC-10-Temp 41.00 125.00 105.00 ok PMIC-11-Temp 43.00 125.00 105.00 ok PMIC-12-Temp 43.00 125.00 105.00 ok PMIC-13-Temp 39.00 125.00 105.00 ok PMIC-14-Temp 36.00 125.00 105.00 ok PMIC-15-Temp 36.00 125.00 105.00 ok PMIC-16-Temp 36.00 125.00 105.00 ok PMIC-17-Temp 30.00 120.00 ok PMIC-18-Temp 30.00 115.00 105.00 ok SODIMM-1-Temp 29.00 95.00 85.00 ok SODIMM-2-Temp 29.38 95.00 85.00 ok admin@rosalind-eb2-2102:~$ admin@nvos:~$ nv show platform environment temperature CPU-Pack-Temp operational ------- ----------- state ok current 30.50 max 95.00 crit 100.00
REST API	GET https://<ip>/nvue_v1/platform/environment/temperature GET https://<ip>/nvue_v1/platform/environment/temperature/<sensor-id>
Related Commands	nv show platform environment
Notes	Note the quotes needed for sensor ID containing a space.

6.5.3.6 nv show platform environment voltage

	nv show platform environment voltage Displays a table with all voltage sensors located on the chassis.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show platform environment voltage Name Actual (V) Maximum (V) Minimum (V) State ---------- ---------- ---------- ---------- ----- HSC-VinDC-In 54.35 88.72 0.03 ok HSC-VinDC-Out 54.39 0.00 ok PDB-1-Conv-In-1 54.20 61.50 38.00 ok PDB-1-Conv-Out-1 12.00 14.10 10.00 ok PDB-2-Conv-In-1 53.70 61.50 38.00 ok PDB-2-Conv-Out-1 11.99 14.10 10.00 ok PMIC-1-ASIC1-VDD-Out-1 0.70 0.98 0.35 ok PMIC-1-PVIN1-VDD-ASIC1-In-1 12.06 16.00 8.00 ok PMIC-2-ASIC1-AVDD-PL0-Out-1 1.05 1.36 0.55 ok PMIC-2-ASIC1-DVDD-PL0-Out-2 0.74 0.90 0.25 ok PMIC-2-PVIN1-AVDD-DVDD-ASIC1-In-1 12.00 16.00 8.00 ok PMIC-3-ASIC1-AVDD-PL1-Out-1 1.05 1.36 0.55 ok PMIC-3-ASIC1-DVDD-PL1-Out-2 0.74 0.90 0.25 ok PMIC-3-PVIN1-AVDD-DVDD-ASIC1-In-1 12.06 16.00 8.00 ok PMIC-4-ASIC1-AVCC-PL0-PL1-Out-1 1.80 2.00 1.30 ok PMIC-4-ASIC1-HVDD-PL0-PL1-Out-2 1.20 1.50 0.80 ok PMIC-4-PVIN1-AVCC-HVDD-ASIC1-In-1 12.00 16.00 8.00 ok PMIC-5-ASIC2-VDD-Out-1 0.70 0.98 0.35 ok PMIC-5-PVIN1-VDD-ASIC2-In-1 12.06 16.00 8.00 ok PMIC-6-ASIC2-AVDD-PL0-Out-1 1.05 1.36 0.55 ok PMIC-6-ASIC2-DVDD-PL0-Out-2 0.74 0.90 0.25 ok PMIC-6-PVIN1-AVDD-DVDD-ASIC2-In-1 12.06 16.00 8.00 ok PMIC-7-ASIC2-AVDD-PL1-Out-1 1.05 1.36 0.55 ok PMIC-7-ASIC2-DVDD-PL1-Out-2 0.74 0.90 0.25 ok PMIC-7-PVIN1-AVDD-DVDD-ASIC2-In-1 12.06 16.00 8.00 ok PMIC-8-ASIC2-AVCC-PL0-PL1-Out-1 1.80 2.00 1.30 ok PMIC-8-ASIC2-HVDD-PL0-PL1-Out-2 1.20 1.50 0.80 ok PMIC-8-PVIN1-AVCC-HVDD-ASIC2-In-1 12.06 16.00 8.00 ok PMIC-9-ASIC3-VDD-Out-1 0.71 0.98 0.35 ok PMIC-9-PVIN1-VDD-ASIC3-In-1 12.06 16.00 8.00 ok PMIC-10-ASIC3-AVDD-PL0-Out-1 1.05 1.36 0.55 ok PMIC-10-ASIC3-DVDD-PL0-Out-2 0.74 0.90 0.25 ok PMIC-10-PVIN1-AVDD-DVDD-ASIC3-In-1 12.12 16.00 8.00 ok PMIC-11-ASIC3-AVDD-PL1-Out-1 1.05 1.36 0.55 ok PMIC-11-ASIC3-DVDD-PL1-Out-2 0.74 0.90 0.25 ok PMIC-11-PVIN1-AVDD-DVDD-ASIC3-In-1 12.09 16.00 8.00 ok PMIC-12-ASIC3-AVCC-PL0-PL1-Out-1 1.80 2.00 1.30 ok PMIC-12-ASIC3-HVDD-PL0-PL1-Out-2 1.20 1.50 0.80 ok PMIC-12-PVIN1-AVCC-HVDD-ASIC3-In-1 12.06 16.00 8.00 ok PMIC-13-ASIC4-VDD-Out-1 0.70 0.98 0.35 ok PMIC-13-PVIN1-VDD-ASIC4-In-1 12.06 16.00 8.00 ok PMIC-14-ASIC4-AVDD-PL0-Out-1 1.05 1.36 0.55 ok PMIC-14-ASIC4-DVDD-PL0-Out-2 0.74 0.90 0.25 ok PMIC-14-PVIN1-AVDD-DVDD-ASIC4-In-1 12.06 16.00 8.00 ok PMIC-15-ASIC4-AVDD-PL1-Out-1 1.05 1.36 0.55 ok PMIC-15-ASIC4-DVDD-PL1-Out-2 0.74 0.90 0.25 ok PMIC-15-PVIN1-AVDD-DVDD-ASIC4-In-1 12.06 16.00 8.00 ok PMIC-16-ASIC4-AVCC-PL0-PL1-Out-1 1.80 2.00 1.30 ok PMIC-16-ASIC4-HVDD-PL0-PL1-Out-2 1.20 1.50 0.80 ok PMIC-16-PVIN1-AVCC-HVDD-ASIC4-In-1 12.06 16.00 8.00 ok PMIC-17-12V-MAIN-In-1 12.06 ok PMIC-17-CPU-Out-1 1.03 1.23 0.61 ok PMIC-17-SOC-Out-2 0.96 1.16 0.53 ok PMIC-18-COMEX-VDD-MEM-In-1 12.00 16.00 ok PMIC-18-COMEX-VDD-MEM-Out-1 1.20 1.42 0.90 ok
REST API	GET https://<ip>/nvue_v1/platform/environment/voltage
Related Commands	nv show platform environment voltage
Notes	

6.5.3.7 nv show platform environment voltage sensor

	nv show platform environment voltage <sensor-id> Display voltage of a specific sensor.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show platform environment voltage PMIC-1-12V-VDD-ASIC1-In-1 operational ------ ---------- state ok actual 13.44 max 16.00 min 0.03

REST API	GET https://<ip>/nvue_v1/platform/environment/voltage/{sensor-id}
Related Commands	nv show platform environment voltage
Notes	This command is supported only on QM9700 platforms

6.5.3.8 nv action turn-on/turn-off platform environment led UID

	nv action turn-on platform environment led UID nv action turn-off platform environment led UID Turns on/off the LED UID.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv action turn-on platform environment led UID`	
REST API	POST https://<ip>/nvue_v1/platform/environment/led/UID	
Related Commands		
Notes		

6.6 Software

NVOS provides tools to display the software packages installed on the system.

6.6.1 Software Commands

- Software Commands

6.6.2 Software Commands

6.6.2.1 nv show platform software

	nv show platform software Display detailed platform software information installed.	
Syntax Description	N/A	
Default	N/A	

History	25.02.5002
Example	admin@nvos:~$ nv show platform software Installed Software ========== Installed Software Description Package Version ---------- ---------- ------- ------- acpi displays information on ACPI devices acpi 1.7-1.2 ...
REST API	GET https://<ip>/nvue_v1/platform/software
Related Commands	
Notes	

6.6.2.2 nv show platform software installed

	nv show platform software installed Displays the software packages installed in the system.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show platform software installed Installed Software Description Package Version ---------- ---------- ---------- ------- acpi displays information on ACPI devices acpi 1.7-1.2 acl access control list - utilities acl 2.2.53-10 adduser add and remove users and groups adduser 3.118 apparmor user-space parser utility for AppArmor apparmor 2.13.6-10 apt commandline package manager apt 2.2.4 apt-transport-https transitional package for https support apt-transport-https 2.2.4 audisp-tacplus audisp module for TACACS+ accounting audisp-tacplus 1.0.2 auditd User space tools for security auditing auditd 1:3.0-2 base-files Debian base system miscellaneous files base-files 11.1+deb11u3 base-passwd Debian base system master password and group files base-passwd 3.5.51 ...	
REST API	GET https://<ip>/nvue_v1/platform/software/installed GET https://<ip>/nvue_v1/platform/software/installed/{installed-id}	
Related Commands	nv show platform	
Notes		

6.6.2.3 nv show platform software installed id

	show platform software installed <installed-id> Display the information for a specific installed software package.

Syntax Description	installed-id	Package name (string)
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show platform software installed acl
	operational
	----------- ------------------------------
	package acl
	version 2.2.53-10
	description access control list - utilities</pre>	

REST API	GET https://<ip>/nvue_v1/platform/software/installed/{installed-id}	
Related Commands	nv show platform	
Notes		

6.7 Transceiver

NVOS offers advanced features for managing and monitoring platform transceivers, ensuring seamless operation and optimal network performance. These features provide users with comprehensive tools for overseeing transceiver status, resetting devices, and managing firmware updates.

6.7.1 Key Functionalities

- Transceiver Status Monitoring: View detailed information about installed transceivers, including their operational state and specifications.
- Transceiver Reset: Perform targeted resets on specific transceivers to resolve issues or refresh their operational state.
- Firmware Management:
  - Install firmware files to update transceivers with the latest features and fixes
  - View the current firmware version of transceivers
  - List and inspect available firmware files

⚠ Using NVIDIA cables is mandatory.

6.7.2 How to Install Transceiver Firmware

Follow the steps below for installing the transceiver firmware.

1. Fetch the firmware file.

```
admin@nvos:~$ nv action fetch platform firmware transceiver scp://username[:password]@hostname/path/
46_120_10010_dev_signed.bin
Password:
Action executing ...
File fetched successfully
Action succeeded
```

2. Install a transceiver firmware.

```
admin@nvos:~$ nv action install platform transceiver sw1 firmware files 46_120_10010_dev_signed.bin
Action executing ... 100%
Installed FW version: 46.120.10010
Action succeeded
```

6.7.3 Transceiver Commands

- Transceiver Commands

6.7.4 Transceiver Commands

6.7.4.1 nv show platform transceiver

	nv show platform transceiver [<transceiver-id>] [detail] Display the status of all transceivers as a table. Includes fields such as cable-type, cable-length, vendor-rev, and identifier.	
Syntax Description	transceiver-id	Display the status of a single transceiver (with the same fields as the general command).
	detail	Show detailed information for all all transceivers.
Default	N/A	
History	25.02.2000 25.02.3000 Updated command output 25.02.5002 Added detail option 25.03.10xx Added supported-cable-ib-speed field to command output	

Example

```
admin@nvos:~$ nv show platform transceiver
Transceiver  Identifier                      Vendor name  Vendor PN         Vendor SN
Vendor revision  FW Version
-----------  -----------------------------  -----------  ----------------  -------------
---------------  -----------
fnm1         OSFP 8X Pluggable Transceiver  NVIDIA       980-9IAHB-00XM0N  MT2445FT25911  44
70.230.1023
sw9          OSFP 8X Pluggable Transceiver  NVIDIA       980-9IAHB-00XM0N  MT2443FT01087  44
70.230.1023
sw10         OSFP 8X Pluggable Transceiver  NVIDIA       980-9IAHB-00XM0N  MT2443FT01092  44
70.230.1023
sw61         OSFP 8X Pluggable Transceiver  NVIDIA       980-9IAHB-00XM0N  MT2443FT01104  44
70.230.1023
sw67         OSFP 8X Pluggable Transceiver  NVIDIA       980-9IAHB-00XM0N  MT2443FT01035  44
70.230.1023
sw68         OSFP 8X Pluggable Transceiver  NVIDIA       980-9IAHB-00XM0N  MT2443FT01097  44
70.230.1023
```

```
admin@nvos:~$ nv show platform transceiver detail

sw1:
  diagnostics-status     : Non present module
  status                 : Removed
  error-status           : N/A

sw2:
  diagnostics-status     : Non present module
  status                 : Removed
  error-status           : N/A

sw3:
  cable-type             : Optical module
  supported-cable-length : 500m SMF
  diagnostics-status     : Diagnostic Data Available
  status                 : Inserted
  error-status           : N/A
  vendor-data-code       : 2025-01-19
  identifier             : OSFP 8X Pluggable Transceiver
  vendor-rev             : 11
  vendor-name            : NVIDIA
  vendor-pn              : 980-9IAHA-00XM0L
  vendor-sn              : MT2503NS00754
  fw-version             : 130.1.0
  temperature:
    temperature         : 53.00 C
    high-alarm-threshold: 80.00 C
    low-alarm-threshold : -10.00 C
  voltage:
    voltage             : 3.25 V
    high-alarm-threshold: 3.50 V
    low-alarm-threshold : 3.10 V
  channel:
    channel-1:
      rx-power:
        power           : 2.64 mW / 4.22 dBm
        high-alarm-thresh: 7.00 dBm
        low-alarm-thresh : -9.83 dBm
      tx-power:
        power           : 2.20 mW / 3.42 dBm
        high-alarm-thresh: 7.00 dBm
        low-alarm-thresh : -6.82 dBm
      tx-bias-current:
        current         : 197.60 mA
        high-alarm-thresh: 480.00 mA
        low-alarm-thresh : 40.00 mA
      rx-cdr-lol    : False
      rx-los        : False
      tx-ad-eq-fault: False
      tx-cdr-lol    : False
      tx-los        : False
      tx-fault      : False
...
.....
```

```
admin@nvos:~$ nv show platform transceiver sw9

cable-type              : Optical module
supported-cable-length  : 500m SMF
supported-cable-ib-speed : sdr,fdr,edr,hdr,ndr
diagnostics-status      : Diagnostic Data Available
status                  : Inserted
error-status            : N/A
vendor-data-code        : 2024-10-21
identifier              : OSFP 8X Pluggable Transceiver
vendor-rev              : 44
vendor-name             : NVIDIA
vendor-pn               : 980-9IAHB-00XM0N
vendor-sn               : MT2443FT01087
fw-version              : 70.230.1023
temperature:
  temperature         : 53.00 C
  high-alarm-threshold: 75.00 C
  low-alarm-threshold : -10.00 C
voltage:
```

```
    voltage             : 3.29 V
    high-alarm-threshold: 3.50 V
    low-alarm-threshold : 3.10 V
  mod-fw-fault           : False
  dp-fw-fault            : False
  channel:
    channel-1:
      rx-power:
          power           : 3.22 mW / 5.08 dBm
          high-alarm-thresh: 6.00 dBm
          low-alarm-thresh : -2.01 dBm
      tx-power:
          power           : 2.20 mW / 3.42 dBm
          high-alarm-thresh: 6.00 dBm
          low-alarm-thresh : 0.00 dBm
      tx-bias-current:
          current         : 103.05 mA
          high-alarm-thresh: 149.80 mA
          low-alarm-thresh : 50.00 mA
      rx-cdr-lol    : False
      rx-los        : False
      tx-ad-eq-fault: False
      tx-cdr-lol    : False
      tx-los        : False
      tx-fault      : False

    channel-2:
      rx-power:
          power           : 2.60 mW / 4.15 dBm
          high-alarm-thresh: 6.00 dBm
          low-alarm-thresh : -2.01 dBm
      tx-power:
          power           : 2.08 mW / 3.18 dBm
          high-alarm-thresh: 6.00 dBm
          low-alarm-thresh : 0.00 dBm
      tx-bias-current:
          current         : 125.83 mA
          high-alarm-thresh: 149.80 mA
          low-alarm-thresh : 50.00 mA
      rx-cdr-lol    : False
      rx-los        : False
      tx-ad-eq-fault: False
      tx-cdr-lol    : False
      tx-los        : False
      tx-fault      : False

  ...

    channel-8:
      rx-power:
          power           : 2.40 mW / 3.80 dBm
          high-alarm-thresh: 6.00 dBm
          low-alarm-thresh : -2.01 dBm
      tx-power:
          power           : 2.21 mW / 3.44 dBm
          high-alarm-thresh: 6.00 dBm
          low-alarm-thresh : 0.00 dBm
      tx-bias-current:
          current         : 107.92 mA
          high-alarm-thresh: 149.80 mA
          low-alarm-thresh : 50.00 mA
      rx-cdr-lol    : False
      rx-los        : False
      tx-ad-eq-fault: False
      tx-cdr-lol    : False
      tx-los        : False
      tx-fault      : False
```


REST API	GET https://<ip>/nvue_v1/platform/transceiver/ GET https://<ip>/nvue_v1/platform/transceiver/{transceiver-id}
Related Commands	Transceiver Commands#nv show platform
Notes	

6.7.4.2 nv action reset platform transceiver id

	nv action reset platform transceiver <transceiver-id> Reset specific transceiver module	
Syntax Description	transceiver-id	The name of the transceiver
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv action reset platform transceiver sw1 Action executing ... Resetting module sw1 ... OK Action succeeded admin@nvos:~$ nv action reset platform transceiver sw1 Action executing ... Resetting module sw1 ... Failed Action succeeded
REST API	POST https://<ip>/nvue_v1/platform/transceiver/{transceiver-id}
Related Commands	nv show platform transceiver <transceiver-id> firmware
Notes	

6.7.4.3 nv action install platform transceiver firmware files

	nv action install platform transceiver <transceiver-id> firmware files <file-name> Install firmware image file on one or more transceiver modules.	
Syntax Description	transceiver-id	The name of the transceiver (e.g. `sw1`)
	file-name	The name of the FW image file
Default	N/A	
History		
Example	admin@nvos:~$ nv action install platform transceiver sw1 firmware files sec_issu_46_120_10010_dev_signed.bin Action executing ... 100% Installed FW version: 46.120.10010 Action succeeded admin@nvos:~$ nv action install platform transceiver sw1-2 firmware files sec_issu_46_120_10010_dev_signed.bin Action executing ... Burning firmware on 2 modules: sw1, sw2. This can take several minutes and cannot be canceled once started - please wait for it to complete. Action executing ... 1% sw1: Installing firmware: sec_issu_46_120_10010_dev_signed.bin Action executing ... 50% sw1: Installed FW: 46.120.10010 Action executing ... 51% sw2: Installing firmware: sec_issu_46_120_10010_dev_signed.bin Action executing ... 100% sw2: Installed FW: 46.120.10010 Firmware burn complete: 2 burned, 0 already up-to-date, 0 failed Action succeeded admin@nvos:~$ nv action install platform transceiver sw5 firmware files sec_issu_46_120_10011_dev_signed.bin Error: Action failed with the following issue: FW update is not supported for this module: no FW mgmt data is found	
REST API	POST https://<ip>/nvue_v1/platform/transceiver/{transceiver-id}/firmware/files/{file-name}	
Related Commands	nv show platform transceiver <transceiver-id> firmware nv show platform transceiver <transceiver-id> firmware files	
Notes	Burning new firmware version is not supported for copper cables.	

6.7.4.4 nv show platform transceiver firmware

	nv show platform transceiver <transceiver-id> firmware Display firmware information for specific transceiver module	
Syntax Description	transceiver-id	The name of the transceiver

Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show platform transceiver sw1 firmware operational applied actual-firmware 46.140.1013 fw-upgrade-status N/A fw-upgrade-error-msg N/A admin@nvos:~$ nv show platform transceiver sw2 firmware operational applied actual-firmware 47.120.10013 fw-upgrade-status OK fw-upgrade-error-msg N/A
REST API	GET https://<ip>/nvue_v1/platform/transceiver/{transceiver-id}/firmware
Related Commands	nv action install platform transceiver <transceiver-id> firmware files {file-name} nv action reset platform transceiver <transceiver-id>
Notes	

6.7.4.5 nv show platform transceiver firmware files

	nv show platform transceiver <transceiver-id> firmware files Display available firmware files for transceiver module	
Syntax Description	transceiver-id	The name of the transceiver
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show platform transceiver sw3 firmware files Available firmware files File path fw1.bin /host/fw-images/modules/fw1.bin fw2.bin /host/fw-images/modules/fw2.bin	
REST API	GET https://<ip>/nvue_v1/platform/transceiver/{transceiver-id}/firmware/files	
Related Commands	nv action install platform transceiver <transceiver-id> firmware files {file-name} nv action reset platform transceiver <transceiver-id> nv show platform transceiver <transceiver-id> firmware	
Notes		

6.7.4.6 nv show platform transceiver firmware files name

	nv show platform transceiver <transceiver-id> firmware files <file-name> Display specific firmware file for transceiver module	
Syntax Description	transceiver-id	The name of the transceiver
	file-name	The name of the FW image file
Default	N/A	
History	25.02.2002	

Example	 ``` admin@nvos:~$ nv show platform transceiver sw3 firmware files fw1.bin Available firmware files File path ------------------------ ---------------------- fw1.bin /host/fw-images/modules/fw1.bin ```
REST API	GET https://<ip>/nvue_v1/platform/transceiver/{transceiver-id}/firmware/files/{file-name}
Related Commands	nv action install platform transceiver <transceiver-id> firmware files {file-name} nv action reset platform transceiver <transceiver-id> nv show platform transceiver <transceiver-id> firmware
Notes	

7 Configuration Management

NVOS provides commands to efficiently manage and apply configurations on the system. These commands allow users to create, view, modify, compare, and save configurations to ensure consistent and accurate network setup. Whether user is making incremental updates, examining configuration differences, or saving changes for persistence across reboots, these tools offer flexibility and control over the system's operational state.

7.1 Managing Configurations as Revisions

NVOS utilizes a revision-based method to manage system configurations effectively. Each configuration change creates a unique revision, providing a clear and structured way to track, compare, and revert modifications.

- Creating Revisions:
  - A new revision is automatically generated whenever user set a configuration using `nv set *`.
  - All subsequent commands will be attached to last created revision.
  - Revision can be applied on system using `nv config apply`.
  - After a revision is applied, it becomes the "applied" revision and cannot be modified further.
  - Each revision includes metadata such as the revision ID, timestamp, the user who made the change, and the method of application (e.g., CLI or API).
- Viewing Revisions:
  - Use the `nv config revision` command to list all stored revisions. This provides a chronological history of all configuration changes made on the system.
- Tracking Changes:
  - The `nv config history` command provides detailed information about revisions, including who made the changes, when they were made, and how they were applied.
- Comparing Revisions:
  - Use `nv config diff <revision> <revision>` to identify differences between two revisions.
  - Compare the current pending configuration with the applied configuration to preview changes before applying them.
- Reverting to a Previous Revision:
  - If a configuration change causes issues, revert to a previous revision using `nv config apply <revision>`.
  - This allows to restore the system to a stable state without manually re-entering configurations.
- Persistence:

    - Changes made to configurations do not persist after a reboot unless explicitly saved using `nv config save`. Saving writes the current configuration to the startup file, making it the default state after a system restart.
- Patch/Replace Configuration:
    - Batch execution support for configuration commands, enabling users to process multiple `nv set` and `nv unset` commands or YAML file as a single atomic transaction through both REST API and CLI interfaces.
    - Use `nv config patch <file>` to update the pending revision with new configurations to or `nv config replace` to replace the pending revision.
    - Commands are submitted in plain-text format and executed as an all-or-nothing operation—if any single command fails, the entire batch is rejected and the revision remains unchanged, ensuring configuration consistency.
    - Supports abbreviated command syntax, Patch security commands when sensitive information is replaced with hashed value.
    - Provides detailed error reporting (including line numbers for invalid commands)
- Verifying Revisions:
    - The `nv config verify` command alows used to perform a dry run to validates configuration changes before applying them
    - Keeps the pending revision unchanged for further edits or actual apply
- Finding Commands and Schema Paths
    - Use `nv config find <search-string>` to search the configuration command tree for commands that match a specified string.
    - Use `nv config lookup <search-path>` to retrieve information about a specified object model or schema path.

7.2 Restoring Factory Default Configuration

Restore to factory default is used when the configuration is corrupted or when there is a need to start the system with default configuration (e.g., when the system is being introduced into a new network).

The "keep" parameter allows to specify different levels of a factory reset. Each option determines which parts of the system configuration, files, and logs are retained or erased. Below are the available options:

1. Full Factory Reset (default): If no specific option is selected, a full reset is performed. This option erases all configuration settings, system files, and log files.

```
admin@nvos:~$ nv action reset system factory-default
```

2. `keep all-config`: This option preserves all system configurations but removes system files and log files. Use this option when you need to retain the configuration but clear any logs or system files.

```
admin@nvos:~$ nv action reset system factory-default keep all-config
```

3. **keep basic** : This option keeps only the basic configuration necessary to maintain system connectivity while removing most of the configuration, system files, and log files. The following settings are retained:
    a. Management interface (eth0, eth1)
    b. Local AAA users and their roles
    c. Password hardening rules
    d. SSH server configuration
    e. DNS server configuration

```
admin@nvos:~$ nv action reset system factory-default keep basic
```

4. **keep only-files** : This option removes all system configuration but retains system and log files. This can be useful for debugging purposes while clearing the configuration.

```
admin@nvos:~$ nv action reset system factory-default keep only-files
```

When doing Reset Factory, the SSD password will be reset to the default password value.

7.2.1 Factory Reset Behavior and Recovery

When you run the following factory reset command, NVOS completes the factory reset process within a few minutes and then reboots the switch automatically:

```
admin@nvos:~$ nv action reset system factory-default
```

A manual reboot is not required during the factory reset process.

Do not power cycle or power off the switch while the factory reset is in progress. If the system is powered off unexpectedly during the reset, the NVOS file system might become corrupted.

If the NVOS file system becomes corrupted, the switch might fail to boot, or it might boot successfully but fail to start critical applications.

To recover the system, use one of the following procedures:

7.2.1.1 If NVOS Does Not Boot

Re-manufacture the switch through the ONIE menu.

7.2.1.2 If NVOS Boots but Critical Applicatins Fail to Start

Run the following command from the Linux shell:

```
docker system prune -a --volumes
```

Then trigger the factory reset again using the **force** option:

```
admin@nvos:~$ nv action reset system factory-default force
```

Wait for the system to complete the reset and reboot automatically.

> When you perform a factory reset, the SSD password is reset to the default password value.

7.3 Configuration Management Commands

7.4 Configuration Management Commands

7.4.1 NVUE Configuration

7.4.1.1 nv config apply

	nv config apply [--assume-yes| --y| --assume-no| --confirm <time>| --confirm-status] Applies the pending configuration to become the applied configuration.	
Syntax Description	--assume-yes | --y | --assume-no	Automatically reply yes/no to all prompts.
	--confirm	Applies the configuration change but you must confirm the applied configuration. If you do not confirm within ten minutes, the configuration rolls back automatically. You can change the default time with the time argument
	--confirm-status	Shows the amount of time left before the automatic rollback.
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv config apply
	Warning: management interface config is changed. If you are connected with it, your conection may be temporary interrupted.
	
	Are you sure? [y/N] y
	applied</pre>	

REST API	PATCH https://<ip>/nvue_v1/revision/{revision-id} *-H 'Content-Type: application/json' -d '{"state": "apply", "auto-prompt": {"ays": "ays_yes"}}'*	
Related Commands		
Notes	NVOS applies but does not save the configuration; the configuration does not persist after a reboot.	

7.4.1.2 nv config detach

	nv config detach Detaches the configuration from the current pending configuration.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	<pre>admin@nvos:~$ nv config detach</pre>
REST API	DELETE https://<ip>/nvue_v1/revision/{revision-id}
Related Commands	
Notes	NVOS names the detached configuration pending and includes a timestamp with extra characters. For example: pending_20210128_212626_4WSY

7.4.1.3 nv config diff

	nv config diff <revision> <revision> | [o commands] | [--expand] Shows differences between configurations, such as the pending configuration and the applied configuration or the detached configuration and the pending configuration.	
Syntax Description	revision	pending configuration / applied configuration / detached configuration / startup configuration
	-o commands	Shows differences between two configuration revisions.
	--expand	Show the configuration in expanded form with no ranges.
Default	N/A	
History	25.02.2002 25.02.6007 Added --expand option	
Example	admin@nvos:~$ nv config diff applied pending - set: interface: swp1: description: Test admin@nvos:~$ nv config diff - set: interface: eth0: description: eth0-desc eth0-1: type: eth eth1: description: eth1-desc admin@nvos:~$ nv config diff --expand - set: interface: eth0: description: eth0-desc type: eth eth1: description: eth1-desc type: eth	
REST API	GET https://<ip>/nvue_v1/<resource>?rev=<rev-A>&diff=<rev-B> GET https://<ip>/nvue_v1/<resource>?rev=<rev-A>&diff=<rev-B>&expand=true	
Related Commands		
Notes		

7.4.1.4 nv config find

	nv config find <string> | [o commands] | [--expand] Finds a portion of the applied configuration according to the provided string.	
Syntax Description	string	Search string
	-o commands	Shows the configurations as commands
	--expand	Show the configuration in expanded form with no ranges.
Default	N/A	

History	25.02.2002 25.02.6007 Added --expand option
Example	admin@nvos:~$ nv config find eth - set: interface: eth0: description: eth0-desc eth0-1: type: eth eth1: description: eth1-desc admin@nvos:~$ nv config find eth --expand - set: interface: eth0: description: eth0-desc type: eth eth1: description: eth1-desc type: eth
REST API	GET https://<ip>/nvue_v1/?rev=applied&filled=False&diff=&search-string=<string> GET https://<ip>/nvue_v1/?rev=applied&filled=False&diff=&search-string=<string>&expand=true
Related Commands	
Notes	

7.4.1.5 nv config history

	nv config history <revision> Shows the apply history for the revision.	
Syntax Description	revision	pending configuration, applied configuration, detached configuration, startup configuration
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv config history 254 Currently pending [rev_id: 435] Rev ID Apply Date Type User Message ------ ------------------------- ---- ----- ---------------------- 254 2025-08-31T10:53:22+03:00 CLI admin Config update by admin	
REST API	GET https://<ip>/nvue_v1/revision/{revision-id} OR GET https://<ip>/nvue_v1/revision	
Related Commands		
Notes		

7.4.1.6 nv config patch

	nv config patch <file> Updates the pending configuration with the specified YAML configuration file or a text file containing NVUE commands.	
Syntax Description	<file>	configuration file

Default	N/A
History	25.02.2002 25.02.6007 Added support for text file containing NVUE commands and added example

Example	Example 1:
	`admin@nvos:~$ nv config patch /deps/nv-02/13/2021.yaml`
	Example 2: Path to text file contains NVUE commands
	`admin@nvos:~$ nv config patch commands.txt`
	`created [rev_id: 1]`
	`admin@nvos:~$ nv config diff`
	`- set:`
	`interface:`
	`acp138:`
	`description: Uplink to spine`
	`link:`
	`state:`
	`up: {}`
	`acp139:`
	`link:`
	`state:`
	`down: {}`
	`admin@nvos:~$ nv config apply -y`
	`applied [rev_id: 1]`

REST API	N/A
Related Commands	nv config apply
Notes	

7.4.1.7 nv config replace

	nv config replace <file> Replaces the pending configuration with the specified YAML configuration file or text file containing NVUE commands.	
Syntax Description	<file>	configuration file
Default	N/A	
History	25.02.2002 25.02.6007 Added support for text file containing NVUE commands and added example	
Example	`admin@nvos:~$ nv config replace /deps/nv-02/13/2021.yaml` `admin@nvos:~$ nv config replace config_backup.txt` `created [rev_id: 6]` `admin@juliet-184-mgmt2:~$ nv config apply` `Warning: current hostname `changed-config` will be replaced with `initial-config` `Are you sure? [y/N] y` `applied [rev_id: 6]`	

REST API	POST https://<ip>/nvue_v1/revision & DELETE https://<ip>/nvue_v1/?rev=<rev-id> & Update the configuration: PATCH https://<ip>/nvue_v1/?rev=<rev-id> Apply changes: PATCH https://<ip>/nvue_v1/revision/{revision-id} -H 'Content-Type: application/json' -d '{"state": "apply", "auto-prompt": {"ays": "ays_yes"}}'
Related Commands	nv config apply
Notes	The replace operation will wipe the default ACL rules for management ports. These rules must be explicitly added to the new configuration file.

7.4.1.8 nv config save

	nv config save Overwrites the startup configuration with the applied configuration. The configuration persists after a reboot.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	`admin@nvos:~$ nv config save`
REST API	PATCH https://<ip>/nvue_v1/revision/{revision-id} *-H 'Content-Type: application/json' -d '{"state": "save", "auto-prompt": {"ays": "ays_yes"}}'*
Related Commands	
Notes	

7.4.1.9 nv config show

	nv config show [o commands] | [--expand] Shows the currently applied configuration in `yaml` format.	
Syntax Description	-o commands	Shows the currently applied configuration commands.
	--expand	Show the configuration in expanded form with no ranges.
Default	N/A	
History	25.02.2002 25.02.6007 Added --expand option	

Example	admin@nvos:~$ nv config show -o commands nv set interface lo type loopback nv set interface swA10p1 link state up nv set interface swA10p1 type ib admin@nvos:~$ nv config show - set: interface: eth0: description: eth0-desc eth0-1: type: eth eth1: description: eth1-desc admin@nvos:~$ nv config show --expand - set: interface: eth0: description: eth0-desc type: eth eth1: description: eth1-desc type: eth
REST API	GET https://<ip>/nvue_v1/?rev=applied&filled=false GET https://<ip>/nvue_v1/?rev=applied&filled=false&expand=true
Related Commands	
Notes	

7.4.1.10 nv show system config files

	nv show system config files Lists the configuration files.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show system config files Available config files File Path ---------- ------- config-file.yaml /host/config_files/config-file.yaml
REST API	GET https://<ip>/nvue_v1/system/config/files
Related Commands	
Notes	

7.4.1.11 nv show system config files

	nv show system config files <file-id> Get a configuration file.	
Syntax Description	file-id	the file name to interact with
Default	N/A	

History	25.02.2002

Example	<pre>- header:
	    model: VX
	    nvue-api-version: nvue_v1
	    rev-id: 1.0
	    version: Cumulus Linux 5.4.0
	- set:
	    system:
	      message:
	        pre-login: Hello
	/etc/nvue.d/config_files/config_file.yaml (END)</pre>

REST API	GET https://<ip>/nvue_v1/system/config/files/{file-name}
Related Commands	
Notes	The command will display the content of the configuration file in 'less' format

7.4.1.12 nv show system config files brief

	nv show system config files <file-id> brief Get a configuration file.	
Syntax Description	file-id	the file name to interact with
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show system config files config-file.yaml brief
	Available config files File path
	---------------------- --
	config-file.yaml /etc/nvue.d/config_files/config-file.yaml</pre>	

REST API	GET https://<ip>/nvue_v1/system/config/files/{file-name}	
Related Commands		
Notes	the command will display the path to the file	

7.4.1.13 nv action export system config

	nv action export system config <file-id> Export configuration file.	
Syntax Description	file-id	the file name to interact with
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv action export system config new_confif_file.yml
	Exporting completed
	Action succeeded</pre>	

REST API	POST https://<ip>/nvue_v1/system/config/files/{file-name}	
Related Commands		
Notes	This command will export the applied configuration a new config file.	

7.4.1.14 nv action rename system config files

	nv action rename system config files <file-id> <new-name> Rename config file.	
Syntax Description	file-id	the file name to interact with
	new-name	the new name to rename to
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action rename system config new_config_file.yml new_name.yaml config file new_config_file.yaml renamed to new_name.yaml Action succeeded	
REST API	POST https://<ip>/nvue_v1/system/config/files/{file-name}	
Related Commands		
Notes		

7.4.1.15 nv action delete system config files

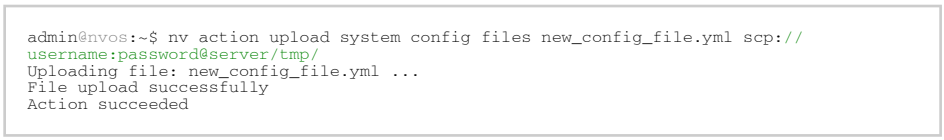
	nv action delete system config files Delete all config files.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv action rename system config new_config_file.yml new_name.yaml config file new_config_file.yaml renamed to new_name.yaml Action succeeded
REST API	POST https://<ip>/nvue_v1/system/config/files
Related Commands	
Notes	

7.4.1.16 nv action delete system config files

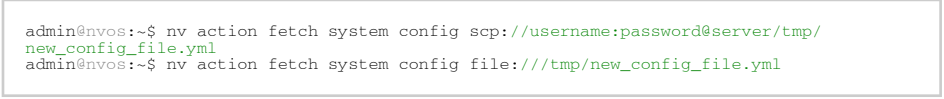
	nv action delete system config files <file-id> Delete config file.	
Syntax Description	file-id	The file name to interact with
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action rename system config new_config_file.yml new_name.yaml config file new_config_file.yaml renamed to new_name.yaml Action succeeded	

REST API	POST https://<ip>/nvue_v1/system/config/files/{file-name}
Related Commands	
Notes	

7.4.1.17 nv action upload system config files

	nv action upload system config files <file-id> <remote-url> Upload config file.	
Syntax Description	file-id	The file name to interact with
	remote-url	Destination image file name Remote url path to upload a file to. Format: [protocol]://username[:password]@hostname/path/filename Supported protocols: SCP, FTP, SFTP, and HTTPS
Default	N/A	
History	25.02.2002 25.02.4002: Added HTTPS support in remote-url	
Example		
REST API	POST https://<ip>/nvue_v1/system/config/files/{file-name}	
Related Commands		
Notes		

7.4.1.18 nv action fetch system config files

	nv action fetch system config files <remote-url> Fetch configuration file.	
Syntax Description	remote-url	• Remote url path to fetch file from. • Format: [protocol]://username[:password]@hostname/path/filename • Supported protocols: SCP, HTTPS, FILE, FTP, and SFTP. • The password must be encoded if it contains special characters and is provided as part of the command.
Default	N/A	
History	25.02.2002	
Example		
REST API	POST https://<ip>/nvue_v1/system/config/files/{file-name}	
Related Commands		

Notes	Alternatively, you can upload a config file from the host machine to the switch. ⚠ Note: you must copy a config to the predefined directory: "/host/config_files/" `user@host:~$ scp <path-to-config-file> <switch-admin-username>@<switch-ip-address>:/host/config_files/<desired-name>.yaml`

7.4.1.19 nv config lookup

	nv config lookup <search-path> [revision options] Identify the configuration hierarchy and retrieve the configuration based on the exact structural path.	
Syntax Description	revision options	Performs a GET request for a specific top-level root path to support path-based lookup across different revisions. Options: `applied`, `pending`, `startup`.
Default	N/A	
History		

Example	`admin@nvos:~$ nv config lookup "system aaa authentication"`
	`- set:`
	`  system:`
	`  aaa:`
	`  authentication:`
	`  restrictions:`
	`  fail-delay: 0`
	`  lockout-state: disabled`
	
	`admin@nvos:~$ nv config lookup "system aaa authentication order" --pending`
	`- set:`
	`  system:`
	`  aaa:`
	`  authentication:`
	`  order:`
	`  - local`
	`  - ldap`	

REST API	GET <https://<ip>>/nvue_v1/?rev=applied&filled=False&search-path=<search-path> GET <https://<ip>>/nvue_v1/?rev=applied&filled=False&search-path=<search-path>&expand=true	
Related Commands		
Notes	Path must be exact and rooted at the top level.	

7.4.2 NVUE Configuration Verify

7.4.2.1 nv config verify

	nv config verify Validate the pending configuration changes without applying them to the system	
Syntax Description	N/A	

Default	N/A
History	25.02.8008
Example	`admin@nvos:~$ nv config verify`
REST API	PATCH https://<ip>/nvue_v1/config/verify
Related Commands	nv config apply
Notes	

7.4.2.2 nv config verify revision

	nv config verify revision <rev_id> Verifies an existing revision by ID. Revision can be integer values, or special revisions like startup or empty.	
Syntax Description	rev_id	The created rev_id
Default	N/A	
History	25.02.8008	
Example	`admin@nvos:~$ nv config verify revision 5`	
REST API	PATCH https://<ip>/nvue_v1/config/verify/revision	
Related Commands	nv config apply	
Notes		

7.4.2.3 nv config verify filename

	nv config verify filename <file.yaml | file.txt> Creates a temporary empty revision, patches the provided YAML or plain text configuration using nv config replace, runs verification.	
Syntax Description	file.yaml | file.txt	Verify configuration from input file.
Default	N/A	
History	25.02.8008	
Example	`admin@nvos:~$ nv config verify filename config.txt`	
REST API	PATCH https://<ip>/nvue_v1/config/verify/filename	
Related Commands	nv config apply	
Notes	During `nv config apply` or `nv config verify`, the system reports only the first encountered error even if multiple issues exist.	

7.4.3 Factory

7.4.3.1 nv action reset system factory-default

	nv action reset system factory-default [force] Clears the system and resets it entirely to its factory state.	
Syntax Description	force	Forces a reboot and configuration reset regardless of the prompt answer
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv action reset system factory-default
	The operation will reset the system configuration and initiate a reboot.
	Type [y] to reset the system configuration and reboot.
	Type [N] to abort.
	
	Do you want to continue? [Y/N] N
	Reset factory aborted by user</pre>	

REST API	POST https://<ip>/nvue_v1/system/	
Related Commands		
Notes	• The command will reboot the system. • The SSD password will be reset to its default value. • This command completes the factory reset and reboots the switch automatically. Do not reboot, power cycle, or power off the switch during the reset. An unexpected interruption can corrupt the NVOS file system. For recovery steps, see Factory Reset Behavior and Recovery.	

8 Telemetry Streaming

The following pages provide telemetry-related information.

8.1 gNMI Streaming

The gRPC Network Management Interface (gNMI) can collect and export system resources, interface, and counter information from NVOS to your gNMI client.

8.1.1 Configure the gNMI Agent using NVUE CLI Commands

The gNMI server feature state can be set over NVOS using simple NVUE CLI commands:

Show command:

```
nvos@switch:~$ nv show system gnmi-server
              operational    applied
------------  -------------  -----------
state         enabled        enabled
certificate   self-signed    self-signed
is-running    yes
version       4.13.0-3000-2
```

Set command:

```
nvos@switch:~$ nv set system gnmi-server state <enabled | disabled>
```

Unset command:

```
nvos@switch:~$ nv unset system gnmi-server state
```

The state is **enabled by default** and the unset command will restore the state to **enabled**, if it is not already.

8.1.2 Supported Subscription Modes

NVOS supports the following gNMI subscription modes:

- STREAM Mode: In this mode, the client subscribes to receive updates whenever there is a change in the telemetry data. This mode is suitable for scenarios where you need real-time notifications of data changes.
- ONCE Mode: This mode retrieves the data once and then terminates the subscription. It's ideal for scenarios where a single snapshot of the data is needed without ongoing updates.
- POLL Mode: In this mode, the client periodically requests data from the server. This mode allows clients to fetch data at defined intervals, providing a balance between real-time and scheduled updates.

Supported stream modes:

- `ON_CHANGE` —When a subscription is defined to be "on change", data updates are only sent when the value of the data item changes.
- `SAMPLE` —This mode allows clients to receive periodic samples of telemetry data at specified intervals. This mode is beneficial for scenarios where continuous streaming of data is not necessary, but periodic updates are required for monitoring and analytics.

Key Parameters for STREAM SAMPLE Mode:

- **sample_interval (mandatory)**: Defines the interval at which samples are sent to the client. This parameter controls the frequency of data transmission.
- **suppress_redundant (optional, default false)**: Determines whether redundant data updates, which have not changed since the last sample, should be suppressed. This helps in reducing unnecessary data transmission and optimizing network usage.
- **heartbeat_interval (optional, default disabled)**: Specifies the interval for sending heartbeat messages to indicate that the connection is still active. Heartbeats help in monitoring the health of the connection and detecting failures.

Subscription Parameter Constraints:

To ensure stable telemetry streaming, the gNMI server enforces the following validation checks. A subscription request will fail if any of these conditions are met:

- **Invalid Mode Constraints:** Neither **sample_interval** nor **heartbeat_interval** can be configured if the subscription mode is not STREAM (e.g., if the mode is ONCE or POLL).
- **Minimum Interval:** The **sample_interval** for a STREAM subscription is below the minimum allowed value of 1 second (1s).
- **Heartbeat Constraints:** The **sample_interval** is strictly greater than the **heartbeat_interval**
- **ON_CHANGE Restrictions:** The **sample_interval** or **suppress_redundant** parameters are configured when the stream mode is set to ON_CHANGE.

8.1.3 gRPC Tunnel (gNMI Dial-out)

The gRPC Tunnel (gNMI Dial-out) feature enables outbound telemetry streaming from the switch to remote collectors or controllers. It allows devices to establish secure, client-initiated connections, making it possible to operate in environments with NAT or restrictive firewall policies where inbound connectivity is not feasible.

The feature provides a configurable framework for managing multiple tunnel endpoints, supporting secure communication using TLS with certificate-based authentication. It integrates with the system's configuration and service infrastructure to handle connection lifecycle, retries, and operational state tracking.

The gRPC Tunnel can be controlled using the following CLI commands:

Display gRPC Tunnel servers:

```
nvos@switch:~$ nv show system grpc-tunnel server
```

Configure new server:

```
nvos@switch:~$ nv set system grpc-tunnel server <test-server> address <address>
nvos@switch:~$ nv set system grpc-tunnel server <test-server> port <port>
```

Set certificate;

```
nvos@switch:~$ nv set system grpc-tunnel server <test-server> certificate my-client-cert
```

8.1.4 Supported Models

8.1.4.1 Models Overview

The NVOS gNMI Model is based on OpenConfig YANG models, extended with NVIDIA-specific augments where required.
It provides a consistent, vendor-neutral telemetry structure while allowing NVIDIA to expose additional InfiniBand, platform, and diagnostic data.

The gNMI YANG models consist of:

- Standard OpenConfig models (baseline support)
- NVIDIA Models (NVOS-specific enrichment)
- Legacy NVIDIA models retained for backward compatibility

8.1.4.2 openconfig-platform

Model	Supported Data
openconfig-interfaces	Base interface configuration, state, and counters: Name, Description, AdminStatus, OperStatus, Enabled, IfIndex, LoopbackMode, and base interface counters (InPkts, OutPkts, InOctets, OutOctets, InUnicastPkts, OutUnicastPkts, InMulticastPkts, OutMulticastPkts, InBroadcastPkts, OutBroadcastPkts, InDiscards, OutDiscards, InErrors, OutErrors), plus InfiniBand-specific interface state (IBSpeed, Speed, IBSubnet, LogicalPortState, PhysicalPortState, MaintenanceState, MTU, MaxSupportedMTUs, SupportedIBSpeeds, SupportedWidths, VLCapabilities, OperationalVL) and InfiniBand port counters (SymbolErrorCounter, XmitWait, RcvErrors, RcvRemotePhyErrors, RcvSwitchRelayErrors, LocalLinkIntegrityErrors, ExcessiveBufferOverrun, LinkErrorRecovery, LinkDowned, QP1Dropped, VL15Dropped and related IB statistics).
openconfig-system	System identity, software, and resource usage: Hostname, BootTime, SoftwareVersion, Location, Contact, RoutingMAC, CPU utilization (aggregate Total/Average), and system memory usage (Physical, Used).
openconfig-platform	Chassis, ASIC, PSU, fan, storage, and other hardware inventory: Component Name, Type, Description, ModelName, PartNo, SerialNo, FirmwareVersion, OperStatus, Temperature, plus component-specific data for fans (Speed, Status), PSUs (Enabled, InputVoltage, InputCurrent, OutputVoltage, OutputCurrent, OutputPower, Status), ASICs (Name, Temperature), chassis/switch (SerialNo, ModelName, PartNo, OperStatus), storage (TotalSize), and platform health (Health Status, LastUnhealthy, UnhealthyCount). And last reboot reason (LastRebootTime, LastRebootReason)
openconfig-platform-transceiver	Optical transceiver module and channel monitoring: module presence and identity (Present, FormFactor, VendorPart, SerialNo), electrical and thermal telemetry (SupplyVoltage, LaserTemperature, module temperature thresholds – Lower, Upper), per-channel optical DOM data (InputPower, OutputPower, LaserBiasCurrent), and per-channel / host-lane status flags (RxCDRLoL, RxLOS, TxCDRLoL, TxLOS, TxFault, TxAdEqFault) and module temperature / voltage alarm flags.
openconfig-platform-healthz	Component health status and history: Status, LastUnhealthy, UnhealthyCount.

8.1.4.3 NVIDIA Models

These models extend OpenConfig to expose NVIDIA-specific telemetry that is not covered by the base OpenConfig schemas.

Model	Supported Data
nvidia-interfaces-infiniband	InfiniBand-specific interface configuration and state: IBSpeed, Speed, IBSubnet, LogicalPortState, PhysicalPortState, MaintenanceState, MTU, MaxSupportedMTUs, SupportedIBSpeeds, SupportedWidths, VLCapabilities, OperationalVL, SpeedNegotiate and related InfiniBand admin/oper fields.

Model	Supported Data
nvidia-interfaces-infiniband-errors-ext	InfiniBand-specific error and status counters: ExcessiveBufferOverrun, LinkErrorRecovery, LinkDowned, LocalLinkIntegrityErrors, RcvErrors, RcvRemotePhyErrors, RcvSwitchRelayErrors, QP1Dropped, VL15Dropped and similar InfiniBand-specific port error counters.
nvidia-system-augments	NVIDIA-specific system metadata: system Location and Contact, plus other NVIDIA system-level extensions modeled as augments to the openconfig-system tree (superseding the legacy platform-general location/contact fields).
nvidia-system-events	Structured system event reporting: EventId, TypeId, Text, Resource, Severity, TimeCreated.
nvidia-if-phy-augments	Enhanced physical-layer diagnostics and BER/FEC telemetry: general PHY and BER state (TimeSinceLastClear, EffectiveErrors, ReceivedBits, SymbolErrors, RawBER, EffectiveBER, SymbolBER, ProfileFECInUse, ZeroHist), per-lane BER and error counters (per-channel RawBER and RawErrors), RS histogram bins (RSCorrectedError counters), link-down statistics (TotalEvents, IntentionalEvents, UnintentionalEvents) and reasons (Local/Remote reason code and status), recovery statistics (LastLogicRecoveryAttempts, LastSerdesEqRecoveryAttempts, TimeBetweenLastTwoRecoveries, TimeInLastLogicRecoveryEvent, TimeInLastSerdesEqRecoveryEvent, TimeSinceLastRecovery, TotalSuccessfulRecoveryEvents), PLR metrics (PLR_BW_LossPercent, PLR_CodesLoss, PLR_RcvCodes, PLR_RcvCodeErr, PLR_RcvUncorrectableCode, PLR_SyncEvents, PLR_XmitCodes, PLR_XmitRetryCodes, PLR_XmitRetryEvents, PLR_XmitRetryEventsWithinTsecMax), and InfiniBand port error and port statistic counters (PortBufferOverrunErrors, PortDLIDMappingErrors, PortInactiveDiscards, PortLocalPhysicalErrors, PortLoopingErrors, PortMalformedPacketErrors, PortNeighborMTUDiscards, PortVLMappingErrors, PortRcvData, PortRcvPkts, PortUnicastRcvPkts, PortUnicastXmitPkts, PortMulticastRcvPkts, PortMulticastXmitPkts, PortXmitData, PortXmitPkts, RQGeneralError, SyncHeaderErrorCounter).
nvidia-platform-integrated-circuit-augments	ASIC power telemetry over standard integrated-circuit model: LongTermAvgPower, ShortTermAvgPower (average power values per monitoring interval on ASIC integrated-circuit power).
nvidia-platform-storage-augments	Switch-local storage utilization: TotalSize for the logical switch storage device
nvidia-platform-transceiver-augments	Transceiver firmware and alarm model: DataPathFirmwareFault, ModuleFirmwareFault, ModuleErrorType and generic alarm state (AlarmStatus, AlarmSeverity, AlarmThreshold) for module temperature and supply voltage, and for channel InputPower, OutputPower and LaserBiasCurrent (replacing legacy module/channel-specific alarm flags).

Model	Supported Data
nvidia-ib-router	InfiniBand (IB) router state model: router enabled status and SWID count, per-subnet operational state (logical-state, physical-state, subnet-prefix, validity, per-ASIC GIDs), and per-subnet interface counters (in/out octets, packets, discards, errors, unicast/multicast packets, excessive-buffer-overrun, xmit-wait, link-downed, link-error-recovery, rcv-switch-relay-errors, rcv-constraints-errors, local-link-integrity-errors).
nvidia-system-image-augments	System Image Partitions' Information: Partition Id, Image build-id, Current and next partitions, Partition description, Partition install disk location, Software Release.
nvidia-platform-reboot	Reboot reason details: last-reboot-details And counters: power-failure, critical-error, user-initiated, total

8.1.4.4 Legacy NVIDIA Models

NVOS exposes a set of **legacy NVIDIA YANG models** for backward compatibility. These models exist **only to support deprecated gNMI xpaths**. All data is available through the Model above, and these models are planned for removal in a future NVOS release.

Model	Supported Data (legacy)
nvidia-platform-general-ext	Legacy platform-wide system and resource information: Contact, Location, NOSVersion, PlatformName, MemoryTotalSize, MemoryUsed, DiskTotalSize, DiskUsed, AmbientTemperature and LeakSensor Id/State.
nvidia-platform-general-ext-versions	Legacy system component firmware inventory: FWVersionBIOS, FWVersionBMC, FWVersionFPGA, FWVersionEROT and FWVersionCPLD / FWVersionSMA entries (per-id version and id).
nvidia-platform-asic	Legacy ASIC-specific telemetry model: ASICName, ASICTemp, LongTermAvgPower, ShortTermAvgPower.
nvidia-if-phy-diag	Legacy PHY diagnostic model: CableProtoCapExt, CoreToPhyLinkProtoEnabled, CoreToPhyLinkWidthEnabled, ETH-AN/IB-PHY/PD/PHY-HST/PHY-Manager FSM and link mode fields, LoopbackMode, FECModeRequest, ProfileFECInUse, EffectiveBER, RawBER, SymbolBER, EffectiveErrors, PhyReceivedBits, SymbolErrors, RS histogram bins (RS_Num_Corr_Err_Bin0–Bin15), PLR_* metrics, InfiniBand port-errors and port-statistics counters, link-down and recovery metrics (LinkDown, IntentionalLinkDownEvents, UnintentionalLinkDownEvents, LinkDownReasonCode/Status Local/Remote, TimeSinceLastClear, TimeBetweenLastTwoRecoveries, TimeInLastLogic/ SerdesEqRecoveryEvent, TimeSinceLastRecovery, TotalSuccessfulRecoveryEvents, ZeroHist), and related PHY diagnostic leaves.
nvidia-platform-transceiver-diag	Legacy transceiver diagnostics model: ModuleOperStatus, DataPathFirmwareFault, ModuleFirmwareFault, ModuleErrorType, module TemperatureHigh/Low Alarm and Warning flags, VccHigh/Low Alarm and Warning flags, and channel-level flags for TxAdEqFault, TxFault, TxCDRLoL, TxLOS, RxCDRLoL, RxLOS.

8.1.4.5 YANG Model Availability

{#000000|The YANG models above are available on the }NVIDIA Enterprise Support Portal{#000000| → Downloads → Switches and Gateways → Switch Software → }QM-3 NVOS InfiniBand → More files.

8.1.4.6 NVOS YANG Package Structure

The NVOS YANG package is provided as a tar archive with the following structure:

```
models/
  ietf                      IETF standard base YANG models
  openconfig                OpenConfig models with NVIDIA Model augments
  nvos                      NVOS-specific OpenConfig augments kept for legacy
                            backward compatibility
  not-supported             Deviation modules that mark non-supported leaves and
                            nodes in the models above
  gnmi-supported-paths.html Reference list of all gNMI-supported paths in this
                            release
```

8.1.5 gNMI Connection and Rate Limiting

The gNMI service enforces limitations on the number of active and incoming gRPC connections to ensure system stability and optimal resource usage.

- Maximum Established Connections:
  The gNMI server supports a maximum of 10 concurrently established gRPC connections at any given time. Once this limit is reached, new connection attempts will be rejected until at least one of the existing connections is terminated.
- Source IP–Based Rate Limiting:
  The gNMI server allows up to 10 concurrent TCP connections from the same source IP address. If additional connection requests are initiated from that IP while the limit is reached, those connection attempts will be dropped automatically. The new connections will only be accepted when the number of active TCP sessions from that IP drops below the configured threshold.
- To enhance the security of gNMI communications, it is strongly recommended to implement mutual TLS (mTLS) authentication together with SPIFFE (Secure Production Identity Framework For Everyone):
  - Mutual TLS (mTLS): Ensures that both client and server authenticate each other using trusted X.509 certificates, thereby preventing unauthorized access and man-in-the-middle attacks.
  - SPIFFE Integration: Leverages SPIFFE IDs to provide consistent, identity-based authentication and authorization across services. This minimizes dependence on static credentials and simplifies certificate management.

8.1.6 Subscription Paths

There is no fixed maximum number of subscription paths.

Each leaf path may create a separate internal subscription. For large models, use container-level or wildcard subscriptions instead of enumerating individual leaf paths.

Recommendations:

- Subscribe to a container or wildcard path that covers multiple leaves.
- Avoid listing hundreds or thousands of individual leaf paths when a single parent or wildcard path is sufficient.

Excessive leaf-level subscriptions may result in high resource utilization, slow subscription setup, or subscription failures under load. If a subscription fails or takes a long time to establish, first reduce the number of paths and use broader subscription targets.

8.1.7 Minimum Sampling Intervals

Two layers determine how often updates are received:

- gNMI protocol interval — The interval specified in the `Subscribe` request.
- On-device metric collection interval — How often NVOS reads the underlying data source.

You may request a faster sampling interval than the device's collection interval. In this case, gNMI updates are still sent at the requested interval, but the underlying metric value is refreshed only at the slower on-device collection rate.

The minimum supported sample interval for all subscription requests is 1 second.

Minimum on-device collection intervals:

Metric category	Examples	Minimum update interval
Interface statistics	Packet and octet counters, operational state, errors	10 seconds
Peer port statistics	Peer port counters and mapping	10 seconds
Platform environment	Temperature, PSU, fan, leak sensors, CPU, memory, and disk	1 second
Transceiver, module level	Status, DOM, and vendor information	1 second
InfiniBand port and router metrics	Port configuration and state, subnet information	1 second
Firmware version	Component firmware versions	30 seconds
Partition and image information	Active and standby image details	30 seconds
Platform counters	ASIC average power and similar platform counters	30 seconds
Events	Event log entries	1 minute
Transceiver channel and lane metrics	Per-channel power, bias current, and similar metrics	1 minute
Device metadata	Hostname, model, and serial number	5 minutes
Reboot information	Reboot count, last reboot time, and cause	5 minutes

Example: If you request firmware version updates every 15 seconds, you receive an update every 15 seconds; however, the underlying data is refreshed only every 30 seconds.

8.1.8 gNMI Client Requests

gNMI client on a host can request capabilities and data from the switch. The examples below use the gNMIc client.

The following example shows a gNMIc **STREAM SAMPLE** mode request for specific Interface data, with a sample interval of 30 seconds, suppress redundant flag enabled, and heartbeat interval of 120 seconds:

```
gnmic -a "IP" --port 9339 --skip-verify subscribe --prefix "interfaces"  --path "/interface[name=sw1p1]"  --target
nvos -u admin -p ***** --mode stream --stream-mode sample --sample-interval 30s --suppress-redundant --heartbeat-
interval 120s
```

The following example shows a gNMIc **STREAM ON-CHANGE** mode request for system events, with an updates-only flag enabled:

```
gnmic -a "IP" --port 9339 --skip-verify subscribe --prefix "/system-events"  --path "" --target nvos -u admin -p
***** --mode stream --stream-mode on-change --updates-only
```

The following example shows a gNMIc **ONCE** mode request and server response for IB interface MTU (-d for debug mode):

```
gnmic -a "IP" --port 9339 --skip-verify subscribe --prefix "interfaces"  --path "/interface[name=sw1p1]/infiniband/
state/mtu" -d --target nvos -u admin -p ***** --mode once
{
  "source": "IP",
  "subscription-name": "default-1709707931",
  "timestamp": 1709707925858795109,
  "time": "2024-03-06T08:52:05.858795109+02:00",
  "prefix": "interfaces/interface[name=sw1p1]",
  "target": "nvos",
  "updates": [
    {
      "Path": "infiniband/state/mtu",
      "values": {
        "infiniband/state/mtu": 256
      }
    }
  ]
}
```

The following example shows a gNMIc **ONCE** request for all supported paths:

```
gnmic -a "IP" --port 9339 --skip-verify subscribe --prefix "/"  --path "" --target nvos -u admin -p ***** --mode
once
```

The following example shows a gNMIc **POLL** mode request and server response for FAN1/1 speed:

```
gnmic -a "IP" --port 9339 --skip-verify subscribe --prefix "components"  --path "component[name=FAN1/1]/fan/state/
speed"  --target nvos -u admin -p *****  --format flat --mode poll
components/component[name=FAN1/1]/fan/state/speed: 33
```

The following example shows a gNMIc **STREAM** mode request for specific system-event "text" leaf with **PROTO** encoding:

```
gnmic -a "IP" --port 9339 --skip-verify subscribe --prefix "system-events"  --path "system-event[event-id=38]/
state/text"  --target nvos -u admin -p ***** --encoding proto --format prototext --mode stream

sync_response:  true

update:  {
  timestamp:  1719295967820127958
```

```
  prefix:  {
    elem:  {
      name:  "system-events"
    }
    elem:  {
      name:  "system-event"
      key:  {
        key:  "event-id"
        value:  "38"
      }
    }
    target:  "nvos"
  }
  update:  {
    path:  {
      elem:  {
        name:  "state"
      }
      elem:  {
        name:  "text"
      }
    }
    val:  {
      string_val:  "Interface admin state is up"
    }
  }
}
```

A list of supported events can be found in the Event Management page.

The following example shows a gRPC curl command to describe the server using gRPC reflection service:

```
docker run fullstorydev/grpcurl -H username:admin -H password:***** -insecure "IP":9339 describe

gnmi.gNMI is a service:
service gNMI {
  rpc Capabilities ( .gnmi.CapabilityRequest ) returns ( .gnmi.CapabilityResponse );
  rpc Get ( .gnmi.GetRequest ) returns ( .gnmi.GetResponse );
  rpc Set ( .gnmi.SetRequest ) returns ( .gnmi.SetResponse );
  rpc Subscribe ( stream .gnmi.SubscribeRequest ) returns ( stream .gnmi.SubscribeResponse );
}
grpc.reflection.v1.ServerReflection is a service:
service ServerReflection {
  rpc ServerReflectionInfo ( stream .grpc.reflection.v1.ServerReflectionRequest ) returns
( stream .grpc.reflection.v1.ServerReflectionResponse );
}
grpc.reflection.v1alpha.ServerReflection is a service:
service ServerReflection {
  rpc ServerReflectionInfo ( stream .grpc.reflection.v1alpha.ServerReflectionRequest ) returns
( stream .grpc.reflection.v1alpha.ServerReflectionResponse );
}
```

The following example shows a gNMIc **ONCE** mode request for all the supported paths:

```
gnmic -a "IP" --port 9339 --skip-verify subscribe --prefix "/"  --path ""  --target nvos -u admin -p ***** --mode
once --format flat
```

The following example shows a gNMIc **Capabilities** request to retrieve the set of capabilities that is supported by the server:

```
gnmic -a "IP" --port 9339 --skip-verify capabilities -u admin -p *****
```

8.1.9 Related Information

- gNMI presentation to IETF
- gNMIc client
- gNMI subscription mode documentation

8.1.10 gNMI Streaming Commands

- gNMI Streaming Commands

8.1.11 gNMI Streaming Commands

- 8.1.11.1 nv show system gnmi-server
- 8.1.11.2 nv show system gnmi-server mtls
- 8.1.11.3 nv set/unset system gnmi-server state
- 8.1.11.4 nv set system gnmi-server certificate
- 8.1.11.5 nv set system gnmi-server mtls ca-certificate
- 8.1.11.6 gNMI Subscriptions
    - 8.1.11.6.1 nv show system gnmi-server status
    - 8.1.11.6.2 nv show system gnmi-server status client
    - 8.1.11.6.3 nv clear system gnmi-server status
- 8.1.11.7 gRPC Tunnel (gNMI Dial-out)
    - 8.1.11.7.1 nv show system grpc-tunnel
    - 8.1.11.7.2 nv show system grpc-tunnel server
    - 8.1.11.7.3 nv show system grpc-tunnel server
    - 8.1.11.7.4 nv show system grpc-tunnel server status
    - 8.1.11.7.5 nv show system grpc-tunnel server status connection
    - 8.1.11.7.6 nv unset system grpc-tunnel
    - 8.1.11.7.7 nv unset system grpc-tunnel server
    - 8.1.11.7.8 nv set/unset system grpc-tunnel server

8.1.11.1 nv show system gnmi-server

	nv show system gnmi-server Displays the gNMI server configured state, actual state, and version.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show system gnmi-server operational applied ---------- ---------- ---------- state enabled enabled certificate self-signed self-signed is-running yes version 4.12.0-PS
REST API	GET https://<ip>/nvue_v1/system/gnmi-server
Related Commands	nv set system gnmi-server state disabled

8.1.11.2 nv show system gnmi-server mtls

	nv show system gnmi-server mtls Return mTLS state information.
Syntax Description	N/A
Default	N/A
History	25.02.4002
Example	admin@nvos:~$ nv show system gnmi-server mtls operational applied -------------- ---------- --------- ca-certificate ca_client ca_client
REST API	GET https://<ip>/nvue_v1/system/gnmi-server/mtls
Related Commands	nv show system gnmi-server
Notes	

8.1.11.3 nv set/unset system gnmi-server state

	nv set system gnmi-server state <enabled | disabled> nv unset system gnmi-server state Sets gNMI server state. The unset form of the command returns the gNMI server state back to its default.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system gnmi-server state disabled admin@nvos:~$ nv unset system gnmi-server state	
REST API	PATCH https://<ip>/nvue_v1/system/gnmi-server DELETE https://<ip>/nvue_v1/system/gnmi-server	
Related Commands	nv unset system gnmi-server state	

8.1.11.4 nv set system gnmi-server certificate

	nv set system gnmi-server certificate {cert id} Set CA certificate for API mTLS connection.	
Syntax Description	certificate	Certificate ID string
Default	self-signed	
History	25.02.2002	
Example	admin@nvos:~$ nv set sys gnmi-server certificate cert_id	
REST API	PATCH https://<ip>/nvue_v1/system/gnmi-server/certificate?rev= Content-Type: application/json {"certificate": "cert_id"}	
Related Commands		
Notes		

8.1.11.5 nv set system gnmi-server mtls ca-certificate

	nv set system gnmi-server mtls ca-certificate <cacert-id> gnmi-server security mTLS information	
Syntax Description	cacert-id	CA certificates [id] for validation of client during mTLS (string)
Default	N/A	
History	25.02.4002	

Example	`admin@nvos:~$ nv set system gnmi-server mtls ca-certificate ca_client`
REST API	PATCH https://<ip>/nvue_v1/system/gnmi-server/mtls/ca-certificate/<arg>
Related Commands	nv show system gnmi-server mtls
Notes	

8.1.11.6 gNMI Subscriptions

8.1.11.6.1 nv show system gnmi-server status

	nv show system gnmi-server status Show the operational state of the gnmi-server.	
Syntax Description	N/A	
Default	N/A	
History	25.02.80xx	

Example	`admin@nvos:~$ nv show system gnmi-server status`
	
	`operational`
	`----------`
	`total-active-subscriptions 2`
	`received-subscription-requests 3`
	`rejected-subscriptions 0`
	`received-capabilities-requests 0`
	`[client] 1`
	`[client]`	

REST API	GET https://<ip>/nvue_v1/system/gnmi-server/status	
Related Commands	nv show system gnmi-server status client	
Notes		

8.1.11.6.2 nv show system gnmi-server status client

	nv show system gnmi-server status client [client-number] Show operational state for a single client connected to the gnmi-server. When no client is specified, the command shows operational state for all gnmi clients connected to the gnmi-server.	
Syntax Description	N/A	
Default	N/A	
History	25.02.80xx	

Example	admin@nvos:~$ nv show system gnmi-server status client client-id active-subscriptions client-address client-port type --------- -------------------- -------------- ----------- -------- 1 1 192.168.200.6 41614 dial-in 2 1 192.168.200.6 50051 dial-out admin@nvos:~$ nv show system gnmi-server status client 1 operational -------------------- ------------- Client-address 192.168.200.6 Client-port 55152 Type dial-in active-subscriptions 1
REST API	GET https://<ip>/nvue_v1/system/gnmi-server/status/client
Related Commands	nv show system gnmi-server status
Notes	

8.1.11.6.3 nv clear system gnmi-server status

	nv clear system gnmi-server status Clear the counters held in the gnmi server operational state.
Syntax Description	N/A
Default	N/A
History	25.02.80xx
Example	admin@nvos:~$ nv clear system gnmi-server status
REST API	DELETE https://<ip>/nvue_v1/system/gnmi-server/status
Related Commands	nv show system gnmi-server status nv show system gnmi-server status client

Notes	• The `clear` command resets the following counters to zero:
	  • `received-subscription-requests`
	  • `rejected-subscriptions`
	  • `received-capabilities-requests`
	• The following values are not affected by the `clear` command, as they change dynamically over time:
	  • `total-active-subscriptions`
	  • Client-related metrics
	• A `clear` command is not supported for `status client`, since there is no persistent state to reset
	• For additional details on behavior and interactions with the gNMI agent, refer to the specification

8.1.11.7 gRPC Tunnel (gNMI Dial-out)

8.1.11.7.1 nv show system grpc-tunnel

	nv show system grpc-tunnel Displays all configured gRPC tunnel servers and their operational status.
Syntax Description	N/A
Default	N/A
History	25.02.80xx
Example	admin@nvos:~$ nv show system grpc-tunnel operational applied -------- ----------- --------- [server] my-server my-server [server] SERVER1 SERVER1
REST API	GET https://<ip>/nvue_v1/system/grpc-tunnel
Related Commands	
Notes	

8.1.11.7.2 nv show system grpc-tunnel server

	nv show system grpc-tunnel server Displays all configured tunnel servers with configuration and connection status.
Syntax Description	N/A
Default	N/A
History	25.02.80xx
Example	admin@nvos:~$ nv show system grpc-tunnel server Address ca-certificate Certificate Port Retry Interval State status.connection.established status.connection.register status.connection.tunnel status.local-address status.local-port status.remote-address status.remote-port Target Name Target Type --------- ----------- -------------- ----------- ----- -------------- -------- ---------- ---------- ---------- ---------- ----- -------------- -------- my-server 10.10.10.10 56001 60 enabled 2026-03-16T18:14:06.824000Z yes yes 10.220.12.118 42036 10.10.10.10 1234 nvos1 gnmi-gnoi SERVER1 10.1.1.10 443 40 disabled 1970-01-01T00:00:00Z no no None 0 None 0 TARGET1 gnmi-gnoi
REST API	GET https://<ip>/nvue_v1/system/grpc-tunnel/server
Related Commands	
Notes	

8.1.11.7.3 nv show system grpc-tunnel server

	nv show system grpc-tunnel server <server-name-id> Displays detailed configuration and operational state for a specific gRPC tunnel server.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
Default	N/A	
History	25.02.80xx	
Example	 admin@nvos:~$ nv show system grpc-tunnel server my-server operational applied ---------- ---------- ---------- state enabled enabled address 10.10.10.10 10.10.10.10 port 56001 56001 target-name nvos1 nvos1 target-type gnmi-gnoi gnmi-gnoi retry-interval 60 60 status local-address 10.220.12.118 local-port 42036 remote-address 10.10.10.10 remote-port 1234 connection established 2026-03-16T18:14:06.824000Z register yes tunnel yes	
REST API	GET https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes		

8.1.11.7.4 nv show system grpc-tunnel server status

	nv show system grpc-tunnel server <server-name-id> status Displays operational status including local and remote connection details.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
Default	N/A	
History	25.02.80xx	
Example	 admin@nvos:~$ nv show system grpc-tunnel server my-server status operational ---------- ---------- local-address 10.220.12.118 local-port 42036 remote-address 10.10.10.10 remote-port 1234 connection established 2026-03-16T18:14:06.824000Z register yes tunnel yes	
REST API	GET https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}/status	
Related Commands		
Notes		

8.1.11.7.5 nv show system grpc-tunnel server status connection

	nv show system grpc-tunnel server <server-name-id> status connection Displays connection-specific details such as registration and tunnel state.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
Default	N/A	
History	25.02.80xx	
Example	admin@nvos:~$ nv show system grpc-tunnel server my-server status connection operational ---------- -------------------------- established 2026-03-16T18:14:06.824000Z register yes tunnel yes	
REST API	GET https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}/status/connection	
Related Commands		
Notes		

8.1.11.7.6 nv unset system grpc-tunnel

	nv unset system grpc-tunnel Remove all gRPC tunnel configuration including all servers and associated settings.	
Syntax Description	N/A	
Default	N/A	
History	25.02.80xx	
Example	admin@nvos:~$ nv unset system grpc-tunnel admin@nvos:~$ nv config apply	
REST API	DELETE https://<ip>/nvue_v1/system/grpc-tunnel	
Related Commands		
Notes	This operation removes all configured tunnel servers and stops all associated tunnel services.	

8.1.11.7.7 nv unset system grpc-tunnel server

	nv unset system grpc-tunnel server Remove all configured gRPC tunnel servers while clearing associated server configurations.	
Syntax Description	N/A	
Default	N/A	
History	25.02.80xx	

Example	`admin@nvos:~$ nv unset system grpc-tunnel server` `admin@nvos:~$ nv config apply`
REST API	DELETE https://<ip>/nvue_v1/system/grpc-tunnel/server
Related Commands	
Notes	All tunnel server instances will be removed and corresponding services will be stopped.

8.1.11.7.8 nv set/unset system grpc-tunnel server

	nv set system grpc-tunnel server <server-name-id> nv unset system grpc-tunnel server <server-name-id> Add new gRPC tunnel server. The unset form of the command removed the gRPC tunnel server.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
Default	N/A	
History	25.02.80xx	
Example	`admin@nvos:~$ nv set system grpc-tunnel server my-server`	
REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes	All tunnel server instances will be removed and corresponding services will be stopped.	

8.1.11.7.9 nv set/unset system grpc-tunnel server address

	nv set system grpc-tunnel server <server-name-id> address <address> nv unset system grpc-tunnel server <server-name-id> address Set IP address or hostname of the gRPC server. The unset form of the command deletes the IP address or hostname.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
	address	Remote tunnel server IP or hostname
Default	N/A	
History	25.02.80xx	
Example	`admin@nvos:~$ nv set system grpc-tunnel server my-server address 10.10.10.10`	
REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes		

8.1.11.7.10 nv set/unset system grpc-tunnel server port

	nv set system grpc-tunnel server <server-name-id> port <port> nv unset system grpc-tunnel server <server-name-id> port Set the port of the gRPC server. The unset form of the command deletes the port.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
	port	TCP port (1–65535)
Default	N/A	
History	25.02.80xx	
Example	`admin@nvos:~$ nv set system grpc-tunnel server my-server port 56001`	
REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes		

8.1.11.7.11 nv set/unset system grpc-tunnel server target-name

	nv set system grpc-tunnel server <server-name-id> target-name <target-name> nv unset system grpc-tunnel server <server-name-id> target-name Set the target name of the gRPC server. The unset form of the command deletes the target name.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
	target-name	Device identity used for registration (e.g., nvos1)
Default	N/A	
History	25.02.80xx	
Example	`admin@nvos:~$ nv set system grpc-tunnel server my-server target-name nvos1`	
REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes		

8.1.11.7.12 nv set/unset system grpc-tunnel server target-type

	nv set system grpc-tunnel server <server-name-id> target-type <target-type> nv unset system grpc-tunnel server <server-name-id> target-type Set the target type of the gRPC server. The unset form of the command deletes the target type.	
Syntax Description	server-name-id	Unique identifier of the tunnel server

	target-type	enum: gnmi-gnoi (only supported value)
Default	N/A	
History	25.02.80xx	
Example	admin@nvos:~$ nv set system grpc-tunnel server my-server target-type gnmi-gnoi	
REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes		

8.1.11.7.13 nv set/unset system grpc-tunnel server state

	nv set system grpc-tunnel server <server-name-id> state <state> nv unset system grpc-tunnel server <server-name-id> state Set the state of the gRPC server. The unset form of the command returns the state of the gRPC server to default.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
	state	enable | disable: Enable or disable tunnel
Default	Enable	
History	25.02.80xx	
Example	admin@nvos:~$ nv set system grpc-tunnel server my-server state enabled	
REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes		

8.1.11.7.14 nv set/unset system grpc-tunnel server retry-interval

	nv set system grpc-tunnel server <server-name-id> retry-interval <retry-interval> nv unset system grpc-tunnel server <server-name-id> retry-interval Set the retry-interval of the gRPC server. The unset form of the command returns the retry-interval of the gRPC server to default.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
	retry-interval	Reconnect interval (10–300 sec)
Default	60	
History	25.02.80xx	
Example	admin@nvos:~$ nv set system grpc-tunnel server my-server retry-interval 60	

REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}
Related Commands	
Notes	

8.1.11.7.15 nv set/unset system grpc-tunnel server certificate

	nv set system grpc-tunnel server <server-name-id> certificate <certificate-id> nv unset system grpc-tunnel server <server-name-id> certificate Set the certificate of the gRPC server. The unset form of the command deletes the certificate.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
	certificate-id	Client TLS certificate ID from NVOS certificate store
Default	N/A	
History	25.02.80xx	

Example	<pre>admin@nvos:~nv set system grpc-tunnel server my-server certificate my-client-cert
	admin@nvos:~nv config apply</pre>	

REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes	Certificate must exist in NVOS certificate store before applying configuration.	

8.1.11.7.16 nv set/unset system grpc-tunnel server ca-certificate

	nv set system grpc-tunnel server <server-name-id> ca-certificate <ca-certificate-id> nv unset system grpc-tunnel server <server-name-id> ca-certificate Set the CA certificate of the gRPC server. The unset form of the command deletes the CA certificate.	
Syntax Description	server-name-id	Unique identifier of the tunnel server
	ca-certificate-id	CA certificate ID used to validate remote server certificate
Default	N/A	
History	25.02.80xx	

Example	<pre>admin@nvos:~nv set system grpc-tunnel server my-server ca-certificate my-ca-cert
	admin@nvos:~nv config apply</pre>	

REST API	PATCH https://<ip>/nvue_v1/system/grpc-tunnel/server/{server-name-id}	
Related Commands		
Notes	CA certificate must be valid and present in NVOS certificate store.	

8.2 SNMP

Simple Network Management Protocol (SNMP) is a network protocol for management and monitoring of network devices.

NVOS supports:

- SNMP versions v1 and v2c
- Standard MIBs
- Query mode only

8.2.1 Standard MIBs

MIB	Standard
RFC1213-MIB	RFC 1213
IF-MIB	RFC 2863
ENTITY-MIB	RFC 2737
ENTITY-SENSOR-MIB	RFC 3433

8.2.2 Configuring SNMP

Activate the SNMP server on the switch by running:

```
admin@nvos:~$ nv set system snmp-server state enabled
admin@nvos:~$ nv set system snmp-server listening-address <IPv4|IPv6|all|all-v6|localhost|localhost-v6>
admin@nvos:~$ nv set system snmp-server readonly-community <community-string>
admin@nvos:~$ nv set system snmp-server system-contact <contact>
admin@nvos:~$ nv set system snmp-server system-location <location>
```

Community strings are case sensitive.

Multiple distinct IPv4/IPv6 addresses may be configured as listening addresses, as long as they are present on the system management interface.

Use all/all-v6 to listen on all available management addresses.

8.2.3 SNMP Commands

- SNMP Commands

8.2.4 SNMP Commands

8.2.4.1 nv show system snmp-server

	nv show system snmp-server Show SNMP server configuration.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system snmp-server operational applied pending ---------- ---------- ---------- [listening-address] all all all [readonly-community] efa1f375d76194fa51a3556a97e641e6 efa1f375d76194fa51a3556a97e641e6 efa1f375d76194fa51a3556a97e641e6 auto-refresh-interval 10 10 10 state enabled enabled enabled
REST API	GET https://<ip>/nvue_v1/system/snmp-server
Related Commands	
Notes	

8.2.4.2 nv show system snmp-server listening-address

	nv show system snmp-server listening-address [addr] Show SNMP server listening addresses configuration.	
Syntax Description	addr	Optional, show info of specific address. If omitted, will show a summary of all listening addresses
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output	

Example	admin@nvos:~$ nv show system snmp-server listening-address port vrf --- ---- --- all 161 switch (config) # nv show system snmp-server listening-address all operational applied pending ---- ---------- ------- ------- port 161 161 161
REST API	GET https://<ip>/nvue_v1/system/snmp-server/listening-address GET https://<ip>/nvue_v1/system/snmp-server/listening-address/{addr}
Related Commands	
Notes	

8.2.4.3 nv show system snmp-server readonly-community

	nv show system snmp-server readonly-community Show SNMP server readonly communities.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system snmp-server readonly-community -------------------------------- efa1f375d76194fa51a3556a97e641e6
REST API	GET https://<ip>/nvue_v1/system/snmp-server/readonly-community
Related Commands	
Notes	

8.2.4.4 nv set system snmp-server state

	nv set system snmp-server state <enabled | disabled> Enable or disable the SNMP server.
Syntax Description	N/A
Default	Disabled
History	25.02.2002
Example	admin@nvos:~$ nv set system snmp-server state enabled
REST API	PATCH https://<ip>/nvue_v1/system/snmp-server
Related Commands	

Notes	At least 1 listening-address and 1 readonly-community must be configured in order to enabled SNMP server.

8.2.4.5 nv set system snmp-server listening-address

	nv set system snmp-server listening-address <addr> [port <port>] Configure a listening-address for the SNMP server.	
Syntax Description	addr	The address to set. Both IPv4 and IPv6 are supported. Special addresses may be specified by designated keywords: all (0.0.0.0), localhost (127.0.0.1), all-v6 ([::]), localhost-v6 ([::1])
	port	Optional, set the port for the listening-address. If omitted, will use the default SNMP port 161.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system snmp-server listening-address 1.1.1.1 port 44444	
REST API	PATCH https://<ip>/nvue_v1/system/snmp-server/listening-address/{addr}	
Related Commands		
Notes	Multiple listening-addresses may be configured.	

8.2.4.6 nv set system snmp-server readonly-community

	nv set system snmp-server readonly-community <community> Configure a readonly community for the SNMP server	
Syntax Description	community	The community to set
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system snmp-server readonly-community community1	
REST API	PATCH https://<ip>/nvue_v1/system/snmp-server/readonly-community/{community}	
Related Commands		
Notes	Up to 4 readonly communities may be configured. The same communities are used for both IPv4 and IPv6.	

8.2.4.7 nv set system snmp-server auto-refresh-interval

	nv set system snmp-server auto-refresh-interval {interval} Set the interval in seconds between SNMP data refreshes	
Syntax Description	interval	The interval to set, in seconds. The allowed range is [5-300]
Default	60	

History	25.02.2002
Example	`admin@nvos:~$ nv set system snmp-server auto-refresh-interval 5`
REST API	PATCH https://<ip>/nvue_v1/system/snmp-server
Related Commands	
Notes	In general, SNMP server refreshes its internal data every time period as set by the auto-refresh-interval configuration. However, sysName is an exception. It reflects the system hostname, and is being refreshed on a less frequent basis, every 12 time periods, as the system hostname is not expected to change very often.

8.2.4.8 nv set system snmp-server system-contact

	nv set system snmp-server system-contact <contact> Set the value for sysContact in MIB-II.	
Syntax Description	contact	The contact to set, up to 255 characters
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system snmp-server system-contact "John Doe"`	
REST API	PATCH https://<ip>/nvue_v1/system/snmp-server	
Related Commands		
Notes		

8.2.4.9 nv set system snmp-server system-location

	nv set system snmp-server system-location <location> Set the value for sysLocation in MIB-II.	
Syntax Description	location	The location to set, up to 255 characters
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system snmp-server system-location "2nd floor cabinet"`	
REST API	PATCH https://<ip>/nvue_v1/system/snmp-server	
Related Commands		
Notes		

9 Monitoring and Diagnostics

9.1 Health Monitoring

NVOS includes a health daemon that is responsible for collecting health events in the system and monitoring various components, including hardware components such as fans, power supply units, leakage sensors, and failing docker containers.

This health daemon runs every 3 seconds, analyzing components. If it detects an issue, it appears in nv show system health and publishes a health event in gNMI.

Health daemon monitors two main items: Service and Hardware

> ⚠ The health daemon begins monitoring the system a few seconds after NVUE is ready to serve. As a result, system health information might not be available for approximately 11 seconds after NVUE becomes ready.

9.1.1 Service Monitoring

The health daemon ensures the continuous operation of essential system services. It monitors:

- Critical Dockers and Services: Verifies that all vital dockers and services are running

9.1.2 Hardware Monitoring

- Leakage Sensors: Detects any potential fluid leaks
- Temperature Sensors: Monitors the temperature of all hardware components to prevent overheating
- Voltage Sensors: Tracks voltage levels across hardware components to ensure proper functionality
- Fan Speeds: Checks the speed of system fans to maintain optimal airflow and cooling
- Power Supply Units (PSUs): Monitors the status of power supply units for stability
- ASIC Health Status: Verifies the health of ASIC components to prevent processing issues
- Disk: Checks the health and free space of the Disk
- CPU: Monitors the CPU utilization and temperature
- Transievers: Monitors the status of the transievers connected to the system

9.1.3 Output Examples

Example output for a healthy system:

Healthy System

```
admin@nvos:~$ nv show system health
           operational  applied
----------  -----------  -------
status      OK
status-led  green


Health issues
=================
No Data
```

Example output for a faulty system:

Faulty System

```
admin@nvos:~$ nv show system health
           operational  applied
----------  -----------  -------
status      Not OK
status-led  amber


Health issues
=================
    Component                    Status information
    ---------------------------  ------------------------------------------------------------
    LEAKAGE-2                    detected leakage
    PMIC 1 Temp                  temperature is too hot, temperature=2008.0, threshold=125.0
```

9.1.4 Health Monitoring Commands

- Health Monitoring Commands

9.1.5 Health Monitoring Commands

9.1.5.1 nv show system health

	nv show system health Show system health status.

Syntax Description	N/A	
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show system health
	            operational applied
	---------- ----------- -------
	status Not OK
	status-led off
	
	Health issues
	================
	    Component Status information
	    --------- ------------------
	    LEAKAGE-1 detected leakage</pre>	

REST API	GET https://<ip>/nvue_v1/system/health	
Related Commands	nv show system health history	
Notes		

9.1.5.2 nv show system health history

	nv show system health history [file-name] Show system health history file.	
Syntax Description	file-name	Show health history files in the system
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show system health history</pre>
	<pre>admin@nvos:~$ nv show system health history files
	
	health history reports File path
	---------------------- -------------------------
	health_history /var/log/health_history
	health_history.1 /var/log/health_history.1</pre>
	<pre>admin@nvos:~$ nv show system health history files health_history</pre>	

REST API	GET https://<ip>/nvue_v1/system/health/history	
Related Commands	nv show system health	
Notes	• When running the command via the CLI, the file open in “Less” • When no file is selected, the default file name that opens is “health_history”	

9.1.5.3 nv show system health component

	nv show system health component Display all system health components with their instances and health state.	
Syntax Description	N/A	
Default	N/A	

History	25.02.7002 25.02.8008 Updated output and added note
Example	admin@nvos:~$ nv show system health component Component Instance State Last Unhealthy Unhealthy Count asic ASIC1 HEALTHY 0 ASIC2 HEALTHY 0 cpu ALL HEALTHY 0 leakage-sensor LEAKAGE-1 HEALTHY 0 LEAKAGE-2 HEALTHY 0 software ALL HEALTHY 0 switch ALL HEALTHY 0
REST API	GET https://<ip>/nvue_v1/system/health/component
Related Commands	nv action clear system health component
Notes	Components with multiple hardware units (asic, fan, transceiver, leakage-sensor, psu) show individual instances. Single instance components (cpu, software, switch) show ALL as the instance name.

9.1.5.4 nv show system health component id

	nv show system health component {component-id} Display system component health for a specific componenet.	
Syntax Description	component-id	Component name (e.g., PSU, Fan, Software)
Default	N/A	
History	25.02.7002	
Example	admin@nvos:~$ nv show system health component Fan state HEALTHY last-unhealthy 2025-01-01 18:00:10 unhealthy-count 1 admin@nvos:~$ $ nv show system health component asic instances Instance State Last Unhealthy Unhealthy Count ASIC1 HEALTHY 0 ASIC2 HEALTHY 0	
REST API	GET https://<ip>/nvue_v1/system/health/component	
Related Commands	nv action clear system health component	
Notes		

9.1.5.5 nv show system health component instance

	nv show system health component <component-id> instance <instance-id> Display detailed health information for a specific component instance.	
Syntax Description	component-id	Component name (e.g., PSU, Fan, Software)
	instance-id	

Default	N/A
History	25.02.8008
Example	admin@nvos:~$ nv show system health component asic instance ASIC1 operational ---------- ---------- state HEALTHY last-unhealthy unhealthy-count 0 admin@nvos:~$ nv show system health component cpu instance ALL operational ---------- ---------- state HEALTHY last-unhealthy unhealthy-count 0
REST API	GET https://<ip>/nvue_v1/system/health/component/<component-id>/instance/<instance-id>
Related Commands	nv action clear system health component
Notes	

9.1.5.6 nv action clear system health component

	nv action clear system health component Clear system components unhealthy information.
Syntax Description	N/A
Default	N/A
History	25.02.7002
Example	admin@nvos:~$ nv action clear system health component Action executing ... Cleared unhealthy information successfully Action succeeded
REST API	POST https://<ip>/nvue_v1/system/health/component
Related Commands	
Notes	The clear command resets only the `last-unhealthy` and `unhealthy-count` fields. The `state` field is not reset.

9.2 Logging

9.2.1 Logging

To display syslog, run the following command:

```
admin@nvos:~$ nv show system log file
```

To display specific syslog file, run the following command:

```
admin@nvos:~$ nv show system log file syslog.1
```

9.2.2 Logging Commands

- Logging Commands

9.2.3 Logging Commands

9.2.3.1 nv show system log

	nv show system log {file {file-name} {follow} {list}} Displays the log file	
Syntax Description	log	Displays the log file in interactive mode , similar to LINUX “less” utility
	file	Displays the contents of the current log
	file-name	Displays an archived log file
	follow	Displays the last few lines of the current log file and then continues to display new lines as they come in until the user hits Ctrl+C, similar to LINUX “tail” utility.
	list	Displays a list of all the available syslog files
Default	N/A	
History	25.02.2002 25.02.3000 Changed "nv show system log files" to "nv show system log file", removed debug-log option, changed output of "nv show system log" to also display log configurations, added "nv show system log file list"	

Example	<pre>admin@nvos:~$ nv show system log
	                operational applied
	------------ ------------ ----------------
	[file] syslog
	[file] syslog.1
	[file] syslog.2.gz
	[component] nvue
	[component] orchagent
	[component] portsyncd
	[component] sai_api_port
	[component] sai_api_switch
	[component] symmetry-manager
	[component] syncd
	rotation
	  frequency daily
	  max-number 20
	  size 10.0
	
	admin@nvos:~$ nv show system log file follow
	May 31 15:59:15.041937 nvos INFO pmon#sensord: PSU-1(L) Temp 3: 36.2 C (min = -0.5 C,
	max = 60.0 C)
	May 31 16:00:01.312936 nvos INFO core_cleanup.py: Cleaning up core files
	May 31 16:00:01.313062 nvos INFO core_cleanup.py: Finished cleaning up core files
	
	admin@nvos:~$ nv show sys log file list
	
	File name File path
	------------ --------------------
	syslog /var/log/syslog
	syslog.1 /var/log/syslog.1
	syslog.2.gz /var/log/syslog.2.gz</pre>

REST API	GET https://<ip>/nvue_v1/system/log GET https://<ip>/nvue_v1/system/log/file GET https://<ip>/nvue_v1/system/log/file/{file-name}
Related Commands	nv set system log component nvue level <level>
Notes	

9.2.3.2 nv show system log component

	nv show system log component <component-name> {file {file-name} {follow} {list}} Displays the log configuration of all the system components.	
Syntax Description	component-name	Displays the log configuration of specific system component
	file	Displays the contents of the current log
	file-name	Displays an archived log file
	follow	Displays the last few lines of the current log file and then continues to display new lines as they come in until the user hits Ctrl+C, similar to LINUX “tail” utility.
	list	Displays a list of all the available syslog files
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output	

Example	admin@nvos:~# nv show system log component Component Level Summary ---------- ------ ------- nvue info orchagent notice portsyncd notice sai_api_port notice sai_api_switch notice symmetry-manager info syncd notice admin@nvos:~$ nv show system log component nvue file list File name File path ---------- ---------- nv-cli.log /var/log/nv-cli.log nvued.log /var/log/nvued.log nvued.log.1 /var/log/nvued.log.1 admin@nvos:~# nv show system log component syncd operational applied ----- ---------- ------- level notice notice
REST API	GET https://<ip>/nvue_v1/system/log/component GET https://<ip>/nvue_v1/system/log/component/{component-name}
Related Commands	nv set system log component nvue level <level>
Notes	

9.2.3.3 nv show system log rotation

	nv show system log rotation Show log rotation criteria configuration.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show system log rotation operational applied ---------- ---------- ------- frequency daily daily max-number 20 20 size 10.0 10.0	
REST API	GET https://<ip>/nvue_v1/system/log/rotation	
Related Commands	nv set system log rotation	
Notes		

9.2.3.4 nv set/unset system log component

	nv set system log component <component-name> level <level> nv unset system log component <component-name> level Set/unset the system component minimum priority level of messages to log	
Syntax Description	component-name	The system component name: nvue, orchagent, portsyncd, sai_api_port, sai_api_switch, syncd
	level	The minimum priority level of messages to log: critical, debug, error, info, notice, warn
Default	notice	

History	25.02.2002
Example	`admin@nvos:~# nv set system log component nvue level debug`
REST API	PATCH https://<ip>/nvue_v1/system/log/component
Related Commands	nv show system log component nv unset system log component level
Notes	To get component-name run "nv show system log component"

9.2.3.5 nv set/unset system log rotation disk-percentage

	nv set system log rotation disk-percentage <percentage> nv unset system log rotation disk-percentage Set/unset the size of the log file to rotate based on disk size percentage.	
Syntax Description	percentage	Percentage value: 0.001–100
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system log rotation disk-percentage 10` `admin@nvos:~$ nv unset system log rotation disk-percentage`	
REST API	PATCH https://<ip>/nvue_v1/system/rotation	
Related Commands	nv show system log rotation	
Notes		

9.2.3.6 nv set/unset system log rotation frequency

	nv set system log rotation frequency <frequency> nv unset system log rotation frequency Set/unset the frequency of the file rotation.	
Syntax Description	frequency	Rotation frequency: daily, weekly, monthly, yearly
Default	daily	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system log rotation frequency weekly` `admin@nvos:~$ nv unset system log rotation frequency`	
REST API	PATCH https://<ip>/nvue_v1/system/rotation	
Related Commands	nv show system log rotation	
Notes		

9.2.3.7 nv set/unset system log rotation max-number

	nv set system log rotation max-number <count> nv unset system log rotation max-number Set/unset the max number of file rotations.	
Syntax Description	count	Number of file rotations before being removed: 0–999999
Default	20	
History	25.02.2002	
Example	admin@nvos:~$ nv set system log rotation max-number 10 admin@nvos:~$ nv unset system log rotation max-number	
REST API	PATCH https://<ip>/nvue_v1/system/rotation	
Related Commands	nv show system log rotation	
Notes		

9.2.3.8 nv set/unset system log rotation size

	nv set system log rotation size <mebibytes> nv unset system log rotation size Set the size of the file to be rotated.	
Syntax Description	mebibytes	The size threshold for a file to be rotated: 0.001–3500 MiB
Default	10	
History	25.02.2002	
Example	admin@nvos:~$ nv set system log rotation size 100.123 admin@nvos:~$ nv unset system log rotation size	
REST API	PATCH https://<ip>/nvue_v1/system/rotation	
Related Commands	nv show system log rotation	
Notes		

9.2.3.9 nv action rotate system log

	nv action rotate system log Force log file rotation. Action to trigger file rotation manually.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action rotate system log Performing syslog log file rotation... Log rotated successfully Action succeeded	

REST API	POST https://<ip>/nvue_v1/system
Related Commands	nv show system log rotation
Notes	

9.3 Remote Logging

To configure remote syslog to send syslog messages to a remote syslog server, follow the steps below.

- Set remote syslog server.

```
admin@nvos:~$ nv set system syslog server <IP address/hostname>
```

- (Optional) Set the destination port of the remote host.

```
admin@nvos:~$ nv set system syslog server <IP address/hostname> port <port-id>
```

- (Optional) Define rules for filtering logs:
  - Create a selector

```
admin@nvos:~$ nv set system syslog selector <selector-id>
```

  - Add filtering rules

```
admin@nvos:~$ nv set system syslog selector <selector-id> severity info
```

  - Attach the selector to the server with priority:

```
admin@nvos:~$ nv set system syslog server <IP address/hostname> selector <priority> selector-id
<selector-id>
```

> If multiple selectors are attached to a single syslog server, they are applied using an OR condition. This means a log line may be sent more than once if it matches more than one selector. To prevent this, it is recommended to use a single selector with combined options (such as severity and filters).

- (Optional) Set the protocol over which to communicate with remote syslog server

```
admin@nvos:~$ nv set system syslog server <IP address/hostname> protocol tcp
```

9.3.1 Remote Logging Commands

- Remote Logging Commands

9.3.2 Remote Logging Commands

9.3.2.1 nv show system syslog

	nv show system syslog Show syslog configuration. It shows the next information: log format, servers list, trap level.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002 25.02.5002 Updated output	

Example	admin@nvos:~$ nv show system syslog operational applied ------ ----------- -------- format standard standard server ======= Servers Vrf Protocol Port Priority Selector-Id -------------------- ------- -------- ---- -------- ----------- 192.0.0.1 mgmt tcp 9080 fe80::202:ff:fe00:2e mgmt udp 514 10 selector-2 20 selector-3 one.one.one.one default tcp 9080 1 selector-3 selector ========== Selectors Severity Program-Name Facility Burst Interval Filter Match Action ---------- -------- ------------ -------- ----- -------- ------ ------------------ ------- selector-1 debug switchd daemon 200 5 1 my_important_id.+$ include selector-2 info ifreload user 10 10 2 ^jobsfailed.+$ exclude selector-3 notice daemon 200 5 1 test exclude 2 restart include $ nv show system syslog operational applied -------- ----------- ----------- trap notice notice format standard standard [server] 10.20.30.40 10.20.30.40 [server] mysyslogsrv mysyslogsrv
REST API	GET https://<ip>/nvue_v1/system/syslog
Related Commands	nv show system log rotation
Notes	

9.3.2.2 nv show system syslog format

	nv show system syslog format Show syslog format.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system syslog format operational applied ------ ----------- -------- format standard standard
REST API	GET https://<ip>/nvue_v1/system/syslog/format
Related Commands	nv show system log rotation
Notes	

9.3.2.3 nv show system syslog server

	nv show system syslog server Show remote syslog servers.

Syntax Description	N/A	
Default	N/A	
History	25.02.2002 25.02.5002 Updated output	
Example	admin@nvos:~$ nv show system syslog server Servers Vrf Protocol Port Priority Selector-Id 40.90.10.5 default tcp 9080 10 my-new-selector 20 my-selector 121.0.10.2 default tcp 514 121.0.10.12 mgmt udp 514 192.0.0.1 red tcp 9080 10 my-new-selector 20 my-selector	
REST API	GET https://<ip>/nvue_v1/system/syslog/server	
Related Commands		
Notes		

9.3.2.4 nv show system syslog server id

	nv show system syslog server <server-id> Show remote syslog server configuration.	
Syntax Description	server-id	Remote syslog server ipv4, or ipv6, or hostname
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output 25.02.5002 Updated output	
Example	admin@nvos:~$ nv show system syslog server 40.90.10.5 operational applied port 9080 9080 protocol tcp tcp vrf default default selector Priority Selector-Id 10 my-new-selector 20 my-selector	
REST API	GET https://<ip>/nvue_v1/system/syslog/server/{server-id}	
Related Commands		
Notes		

9.3.2.5 nv show system syslog selector

	nv show system syslog selector Displays all configured syslog selectors along with severity, filters, and rate-limit data.	
Syntax Description	N/A	
Default	N/A	

History	25.02.5002
Example	admin@nvos:~$ nv show system syslog selector Selectors Severity Program-Name Facility Burst Interval Filter Match Action ---------- -------- ------------ -------- ----- -------- ------ ------------------ ------- selector-1 debug switchd daemon 200 5 1 my_important_id.+$ include selector-2 info ifreload user 10 10 2 ^jobsfailed.+$ exclude selector-3 notice daemon 200 5 1 test exclude 2 restart include
REST API	GET https://<ip>/nvue_v1/system/syslog/selector
Related Commands	
Notes	

9.3.2.6 nv show system syslog selector id

	nv show system syslog selector <selector-id> Shows detailed configuration of a specific selector including severity, facility, and rate limits.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
Default	N/A	
History	25.02.5002	
Example	admin@nvos:~$ nv show system syslog selector selector-3 operational applied ---------- ---------- ------- facility daemon daemon severity notice notice rate-limit ---------- burst 200 200 interval 5 5 filter ======= Priority Action Match -------- ------- ------- 1 exclude test 2 include restart	
REST API	GET https://<ip>/nvue_v1/system/syslog/selector/{selector-id}	
Related Commands		
Notes		

9.3.2.7 nv show system syslog selector id filter

	nv show system syslog selector <selector-id>filter Lists all filter rules under the specified selector, including match patterns and actions.	
Syntax Description	<selector-id>	Specifies the unique reference for a syslog selector.

Default	N/A
History	25.02.5002
Example	admin@nvos:~$ nv show system syslog selector selector-3 filter Priority Action Match -------- ------- ------- 1 exclude test 2 include restart
REST API	GET https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/filter
Related Commands	
Notes	

9.3.2.8 nv show system syslog selector id filter id

	nv show system syslog selector <selector-id> filter <filter-id> Displays detailed operational and applied values for a specific filter within the selector.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
Syntax Description	filter-id	Specifies the unique reference for a filter.
Default	N/A	
History	25.02.5002	
Example	admin@nvos:~$ nv show system syslog selector selector-3 filter 20 operational applied ------ ----------- ------- match test test action exclude exclude	
REST API	GET https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/filter/{priority-id}	
Related Commands		
Notes		

9.3.2.9 nv show system syslog selector id rate-limit

	nv show system syslog selector <selector-id> rate-limit Shows configured burst and interval rate-limit values for the specified selector.	
Syntax Description	selector-id	The number of a specific selector
Default	N/A	
History	25.02.5002	
Example	admin@nvos:~$ nv show system syslog selector selector-2 rate-limit operational applied -------- ----------- ------- burst 10 10 interval 10 10	
REST API	GET https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/rate-limit	

Related Commands	
Notes	

9.3.2.10 nv show system syslog server id selector

	nv show system syslog server <server-id> selector Lists all selectors associated with the specified syslog server along with their priorities.	
Syntax Description	server-id	Remote syslog server ipv4, or ipv6, or hostname
Default	N/A	
History	25.02.5002	
Example	admin@nvos:~$ nv show system syslog server 40.90.10.5 selector Priority Selector-Id -------- --------------- 10 my-new-selector 20 my-selector	
REST API	GET https://<ip>/nvue_v1/system/syslog/server/{server-id}/selector	
Related Commands		
Notes		

9.3.2.11 nv show system syslog server id selector id

	nv show system syslog server <server-id> selector <priority-id> Displays detailed information about a selector mapped to the syslog server for the given priority.	
Syntax Description	server-id	Remote syslog server ipv4, or ipv6, or hostname
	priority-id	Specifies the priority level of a syslog selector
Default	N/A	
History	25.02.5002	
Example	admin@nvos:~$ nv show system syslog server 40.90.10.5 selector 10 operational applied ----------- --------------- --------------- selector-id my-new-selector my-new-selector	
REST API	GET https://<ip>/nvue_v1/system/syslog/server/{server-id}/selector/{priority-id}	
Related Commands		
Notes		

9.3.2.12 nv unset system syslog

	nv unset system syslog Clear syslog global parameters. It completely removes syslog configuration.

Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv unset system syslog`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog	
Related Commands	nv show system log rotation	
Notes		

9.3.2.13 nv set/unset system syslog format

	nv set system syslog format nv unset system syslog format Set the log format. The unset form of the command resets format to default value.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002 25.02.5002 Removed the welf option	
Example	`admin@nvos:~$ nv unset system syslog format`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/format	
Related Commands	nv set system syslog format welf firewall-name	
Notes		

9.3.2.14 nv set/unset system syslog format welf firewall-name

	nv set system format welf firewall-name <name> nv unset system format welf firewall-name Set WELF format firewall name. The unset form of that command clears the firewall name.	
Syntax Description	Name	Log firewall name to include into WELF log format.
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system syslog format welf firewall-name "NVOS switch 2"` `admin@nvos:~$ nv unset system syslog format welf firewall-name`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/format/welf	
Related Commands		
Notes		

9.3.2.15 nv unset system syslog server

	nv unset system syslog server Clear remote syslog servers.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv unset system syslog server`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/server	
Related Commands		
Notes		

9.3.2.16 nv set/unset system syslog server id

	nv set system syslog server {server-id} nv unset system syslog server {server-id} Set new remote syslog server. The unset form of the command clears a specific server.	
Syntax Description	server-id	Remote syslog server ipv4, or ipv6, or hostname
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system syslog server 10.20.30.40` `admin@nvos:~$ nv unset system syslog server 10.20.30.40`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/server/{server-id}	
Related Commands		
Notes		

9.3.2.17 nv set/unset system syslog server id port

	nv set system server <server-id> port <port-number> nv unset system server <server-id> port Set the port over which to communicate with remote syslog server. The unset form of the command returns port to default.	
Syntax Description	server-id	Remote syslog server ipv4, or ipv6, or hostname
	port-number	Port number of the remote syslog server: 1–65535
Default	512	
History	25.02.2002	

Example	admin@nvos:~$ nv set system syslog server 10.20.30.40 port 1234 admin@nvos:~$ nv unset system syslog server 10.20.30.40 port
REST API	PATCH https://<ip>/nvue_v1/system/syslog/server/{server-id}
Related Commands	
Notes	

9.3.2.18 nv set/unset system syslog server id protocol

	nv set system server <server-id> protocol [tcp | udp] nv unset system server <server-id> protocol Set the protocol over which to communicate with remote syslog server. The unset form of the command returns the protocol to default.	
Syntax Description	server-id	Remote syslog server ipv4, or ipv6, or hostname
Default	udp	
History	25.02.2002	
Example	admin@nvos:~$ nv set system syslog server 10.20.30.40 protocol tcp admin@nvos:~$ nv unset system syslog server 10.20.30.40 protocol	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/server/{server-id}	
Related Commands		
Notes		

9.3.2.19 nv set/unset system syslog server id vrf

	nv set system server <server-id> vrf [default | mgmt] nv unset system server <server-id> vrf [default | mgmt] Specify the VRF over which to communicate with the remote server. The unset form of the command returns VRF to default.	
Syntax Description	server-id	Remote syslog server ipv4, or ipv6, or hostname
Default	default	
History	25.02.2002	
Example	admin@nvos:~$ nv set system syslog server 10.20.30.40 vrf mgmt admin@nvos:~$ nv unset system syslog server 10.20.30.40 vrf	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/server/{server-id}	
Related Commands		
Notes		

9.3.2.20 nv unset system syslog selector

	nv unset system syslog selector Clear remote syslog selectors.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv unset system syslog selector`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector	
Related Commands		
Notes		

9.3.2.21 nv set/unset system syslog selector id

	nv set system syslog selector {selector-id} nv unset system syslog server {selector-id} Set new remote syslog selector. The unset form of the command clears a specific selector.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system syslog selector Selector-1` `admin@nvos:~$ nv unset system syslog selector Selector-2`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}	
Related Commands		
Notes		

9.3.2.22 nv set/unset system syslog selector id severity

	nv set system syslog selector <selector-id> severity <level> nv unset system syslog server <selector-id> severity Set the severity level for a specific syslog selector. Unset the severity setting for a specific syslog selector.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
	level	Sets the minimum severity level of logs (debug | info | notice | warn | error | critical)
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv set system syslog selector severity info admin@nvos:~$ nv unset system syslog selector severity
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/severity
Related Commands	
Notes	

9.3.2.23 nv set/unset system syslog selector id program-name

	nv set system syslog selector {selector-id} program-name {value} nv unset system syslog server {selector-id} program-name Set the program name filter for a specific syslog selector. Unset the program name filter for a specific syslog selector.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
	value	Application or Program name
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system syslog selector program-name switchd admin@nvos:~$ nv unset system syslog selector program-name	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/program-name	
Related Commands		
Notes		

9.3.2.24 nv set/unset system syslog selector id facility

	nv set system syslog selector {selector-id} facility {name} nv unset system syslog server {selector-id} facility Set the syslog facility for a specific selector. Unset the facility setting for a specific syslog selector.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
	name	Log source category (auth, authpriv, cron, daemon, kern, lpr, mail, mark, news, security, syslog, user, uucp, local0, local1, local2, local3, local4, local5, local6, local7, null)
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system syslog selector facility daemon admin@nvos:~$ nv unset system syslog selector facility	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/program-name	

Related Commands	
Notes	

9.3.2.25 nv unset system syslog selector id filter

	nv unset system syslog server {selector-id} filter Unset remote syslog filter.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv unset system syslog selector Selector-2 filter`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/filter	
Related Commands		
Notes		

9.3.2.26 nv set system syslog selector id filter id

	nv set system syslog selector {selector-id} filter {filter-id} Set a specific filter for a syslog selector.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
	filter-id	Specifies the unique reference for a filter.
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system syslog selector Selector-1 filter 1` `admin@nvos:~$ nv unset system syslog selector Selector-2 filter 1`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/filter/{filter-id}	
Related Commands		
Notes		

9.3.2.27 nv set system syslog selector id filter id match

	nv set system syslog selector {selector-id} filter {filter-id} match {match-string} Set a match string for a specific filter within a syslog selector	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
	filter-id	Specifies the unique reference for a filter.
	match-string	Condition to match log messages
Default	N/A	

History	25.02.2002
Example	`admin@nvos:~$ nv set system syslog selector Selector-1 filter 1 match my_important_id.+$`
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/filter/{filter-id}/match
Related Commands	
Notes	

9.3.2.28 nv set system syslog selector id filter id action

	nv set system syslog selector {selector-id} filter {filter-id} action {value} Set the action for a specific filter in a syslog selector to either include or exclude matching log entries.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
	filter-id	Specifies the unique reference for a filter.
	value	Action to take on matched log messages (include or exclude)
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system syslog selector Selector-1 filter 1 action include`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/filter/{filter-id}/action	
Related Commands		
Notes		

9.3.2.29 nv unset system syslog selector id rate-limit

	nv unset system syslog server {selector-id} rate-limit The unset form of the command clears a specific rate-limit.	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
Default	N/A	
History	25.02.2002	
Example	`admin@nvos:~$ nv set system syslog selector Selector-1 rate-limit`	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/rate-limit	
Related Commands		
Notes		

9.3.2.30 nv set/unset system syslog selector id rate-limit burst

	nv set system syslog selector {selector-id} rate-limit burst {value} nv unset system syslog server {selector-id} rate-limit burst Sets the burst size for the rate-limiting of syslog messages Unset the rate-limiting burst configuration	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
	value	Maximum number of log messages allowed within the defined interval (1-65535)
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system syslog selector Selector-1 rate-limit burst 1 admin@nvos:~$ nv unset system syslog selector Selector-2 rate-limit burst	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/rate-limit/burst	
Related Commands		
Notes		

9.3.2.31 nv set/unset system syslog selector id rate-limit interval

	nv set system syslog selector {selector-id} rate-limit interval {value} nv unset system syslog server {selector-id} rate-limit interval Sets the time interval for the rate-limiting of syslog messages Unset the rate-limiting interval configuration	
Syntax Description	selector-id	Specifies the unique reference for a syslog selector.
	value	Time duration (in seconds) during which the burst limit is applied (1-65535)
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system syslog selector Selector-1 rate-limit interval 1 admin@nvos:~$ nv unset system syslog selector Selector-2 rate-limit interval	
REST API	PATCH https://<ip>/nvue_v1/system/syslog/selector/{selector-id}/rate-limit/interval	
Related Commands		
Notes		

9.4 Link Diagnostic Per Port

- 9.4.2 Link Diagnostic Commands

When debugging a system, it is important to be able to quickly identify the root of a problem. The Diagnostic commands enables an insight into the physical layer components where the user is able to see information such as a cable status (plugged/unplugged) or if Auto-Negotiation has failed.

9.4.1 PHY Firmware Indication

9.4.1.1 Link Diagnostic Indication

Code	Firmware PHY Indication (0–1023)
0	No issue observed
1	Port is close by command
2–4	Auto Negotiation failure
5–8	Link training failure
9–13	Logical mismatch between link partners
14	Remote fault received
15	Bad Signal integrity
16	Compliance code mismatch (protocol mismatch between cable and port)
17	Bad signal integrity
18	Internal error
19	Internal error
22	Internal error
23	Internal error
24–32	Cable compliance code mismatch (protocol mismatch between cable and port)
34	Speed degradation
35	Speed degradation
38	Auto Negotiation failure
39	Auto Negotiation failure
40	VPI protocol do not match
41	Port is closed, module cannot be set to the enabled rate
42	Bad signal integrity
48	Bad signal integrity
49	Bad signal integrity
50	Internal error
52	Bad signal integrity
55	Internal error

56	module_lanes_frequency_not_synced
57	Signal not detected
60	No partner detected for long time
128	Troubleshooting in process
1023	Information not available
Code	**Firmware Management Issues (1024–2047)**
1024	Cable is unplugged
1025	Long range for non NVIDIA cable/module
1026	Bus stuck (I^2C Data or clock shorted)
1027	Bad/unsupported EEPROM
1028	Part number list
1029	Unsupported cable
1030	Module temperature shutdown
1031	Shorted cable
1032	Power budget exceeded
1033	Management force down the port
1034	Module is disabled by command
1035	System Power is Exceeded therefore the module is powered off
1036	Module's PMD type is not enabled (see PMTPS).
1040	pcie system power slot Exceeded
1042	Module state machine fault
1043–1046	Module's stamping speed degeneration
1047, 1048	Modules DataPath FSM fault
1050–1053	Module Boot Error
1054	Module Forced to Low Power by command
1055	ELS laser fiber is contaminated
1056	ELS laser failure
1057	ELS cable unplugged

9.4.1.2 Link Down Reason Indication

Code	Link Down Reason Indication
0	No_link_down_indication
1	Unknown_reason
2	Hi_SER_or_Hi_BER
3	Block_Lock_loss
4	Alignment_loss

Code	Link Down Reason Indication
5	FEC_sync_loss
6	PLL_lock_loss
7	FIFO_overflow
8	false_SKIP_condition
9	Minor_Error_threshold_exceeded
10	Physical_layer_retransmission_timeout
11	Heartbeat_errors
12	Link_Layer_credit_monitoring_watchdog
13	Link_Layer_integrity_threshold_exceeded
14	Link_Layer_buffer_overrun
15	Down_by_outband_command_with_healthy_link
16	Down_by_outband_command_for_link_with_hi_ber
17	Down_by_inband_command_with_healthy_link
18	Down_by_inband_command_for_link_with_hi_ber
19	Down_by_verification_GW
20	Received_Remote_Fault
21	Received_TS1
22	Down_by_management_command
23	Cable_was_unplugged
24	Cable_access_issue
25	Cable_Thermal_shutdown
26	Current_issue
27	Power_budget
28	Fast_recovery_raw_ber
29	Fast_recovery_effective_ber
30	Fast_recovery_symbol_ber
31	Fast_recovery_credit_watchdog
32	Peer_side_down_to_sleep_state
33	Peer_side_down_to_disable_state
34	Peer_side_down_to_disable_and_port_lock
35	Peer_side_down_due_to_thermal_event
36	Peer_side_down_due_to_force_event
37	Peer_side_down_due_to_reset_event
38	Reset_no_power_cycle
39	Fast_recovery_tx_plr_trigger
40	Down_due_to_HW_force_event

Code	Link Down Reason Indication
41	Down_due_to_thermal_event
42	L1_exit_failure
43	too_many_link_error_recoveries
44	Down_due_to_contain_mode
45	BW_loss_threshold_exceeded

9.4.2 Link Diagnostic Commands

9.4.3 Link Diagnostic Commands

9.4.3.1 nv show interface link diagnostics

	nv show interface <interface-id> link diagnostics Display the link diagnostics information of the given interface.	
Syntax Description	interface-id	Name of the interface to display
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~:~$ nv show interface sw1p1s1 link diagnostics Code Status -------- ---------- 0 No issue was observed	
REST API	GET https://<ip>/nvue_v1/interface/<interface id>/link/diagnostics	
Related Commands	nv show interface --view link-diagnostics	
Notes		

9.4.3.2 nv show interface view link diagnostics

	nv show interface--view link-diagnostics Display the link diagnostics for all the interfaces.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show interface --view link-diagnostics Interface Code Status ---------- ----- -------------------- acp1 2 Negotiation failure acp2 2 Negotiation failure fnm1 2 Negotiation failure fnm2 2 Negotiation failure sw1p1s1 1024 Cable is unplugged sw1p1s2 1024 Cable is unplugged	
REST API	GET https://<ip>/nvue_v1/interface	
Related Commands	nv show interface link diagnostics	
Notes		

9.4.3.3 nv set/unset system lldp

	nv set system lldp [state {enabled,disabled} | tx-interval {number} | tx-hold-multiplier {number}] nv unset system lldp [state {enabled,disabled} | tx-interval {number} | tx-hold-multiplier {number}] Update LLDP global configurations.	
Syntax Description	state	LLDP state configuration.
	tx-interval	Change transmit delay.
	tx-hold-multiplier	TTL of transmitted packets is calculated by multiplying the tx-interval by the given factor.
Default	state	enabled
	tx-interval	30
	tx-hold-multiplier	4
History	25.02.2002	
Example	admin@nvos:~$ nv set system lldp state disabled admin@nvos:~# nv set system lldp state tx-interval 50	
REST API	PATCH https://<ip>/nvue_v1/system/lldp	
Related Commands	nv show system lldp	
Notes		

9.4.3.4 nv show system lldp

	nv show system lldp Display LLDP global configurations.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv show system lldp operational applied state enabled enabled tx-interval 30 30 tx-hold-multiplier 4 4
REST API	GET https://<ip>/nvue_v1/system/lldp
Related Commands	nv set system lldp
Notes	

9.4.3.5 nv show interface lldp

	nv show interface <interface-id> lldp Show preview details of interface neighbor.	
Syntax Description	interface-id	interface name
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show interface eth0 lldp operational applied [neighbor] MTL-S-F2-LAB-ADVG-SW-4-12	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/lldp	
Related Commands	nv show interface {interface-id} lldp neighbor	
Notes	This command is just for mgmt interfaces.	

9.4.3.6 nv show interface lldp neighbor

	nv show interface <interface-id> lldp neighbor Show details of interface neighbor.	
Syntax Description	interface-id	interface name
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show interface eth0 lldp neighbor Neighbor Remote IP Model SW Version Remote Port MTL-S-F2-LAB-ADVG-SW-4-12 10.60.4.12 11
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/lldp/neighbor
Related Commands	nv show interface <interface-id> lldp neighbor <neighbor-id>
Notes	This command is just for mgmt interfaces.

9.4.3.7 nv show interface lldp neighbor id

	nv show interface <interface-id> lldp neighbor <neighbor-id> Show extend details of the neighbor	
Syntax Description	interface-id	Interface name
	neighbor-id	Neighbor name
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show interface eth0 lldp neighbor MTL-S-F2-LAB-ADVG-SW-4-12 operational age 3581 bridge untagged 59 chassis chassis-id 1c:98:ec:45:3e:c0 management-address-ipv4 10.60.4.12 management-address-ipv6 fdfd:fdfd:7:145::1000:4998 system-name MTL-S-F2-LAB-ADVG-SW-4-12 system-description HP J9775A 2530-48G Switch, revision YA.16.11.0015, ROM YA.15.20 ... port ttl 120 name 11 type local description 11 pmd-autoneg [advertised] 0 [advertised] 1 [advertised] 2 mau-oper-type 1000BaseTFD - Four-pair Category 5 UTP, full duplex mode lldp-med device-type	
REST API	GET https://<ip>/nvue_v1/interface/{interface-id}/lldp/neighbor/{neighbor-id}	
Related Commands	nv show interface {interface-id} lldp neighbor	
Notes	This command is just for mgmt interfaces.	

9.4.3.8 nv show interface lldp

	nv show interface lldp Show neighbor details of all interfaces.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show interface lldp Interface Speed Type Neighbor Remote Port --------- ------ -------- ------------------------ ----------- eth0 1G eth MTL-S-F2-LAB-ADVG-SW-4-12 11 ib0 ipoib lo loopback sw1p1 200G ib sw1p2 200G ib sw2p1 ib ... ...
REST API	GET https://<ip>/nvue_v1/interface/lldp
Related Commands	nv show interface {interface-id} lldp
Notes	

9.5 Event Management

The primary objective of incorporating this feature is to address particular actions (such as port disable) or events (like port fast-recovery) and broadcast them in a standardized format along with descriptions. This aims to streamline remote system state monitoring for users.

> In this current release, only the CLI is supported for accessing the list of broadcasted events. However, in the upcoming release, gNMI will be expanded to facilitate the remote publication of each event to clients. To subscribe to gNMI events, see gNMI Streaming section.

9.5.1 Supported Events

The following table presents the supported events with their description.

Resource	Event Description	Severity
System	System fatal state detected	CRITICAL
System	System is not ready—one or more services are not up	CRITICAL
System	System is not ready—one or more services failed	MAJOR
System	Restart all syncd-ibv0 dockers	MAJOR
System	Performing reboot	MAJOR

Resource	Event Description	Severity
System	Health status is not ok	WARNING
System	System is ready	INFORMATIONAL
System	System recovered from fatal state	INFORMATIONAL
System	Health status is ok	INFORMATIONAL
Sensor or service name	<Repeats a message from the system health>	WARNING
Sensor or service name	Hardware component goes back to normal / Service goes back to normal	INFORMATIONAL
Interface name	Interface admin state is {Up/Down}	INFORMATIONAL
Interface name	Interface operational state is {Up/Down}	INFORMATIONAL
Interface name	Fast-recovery error event for trigger {trigger_name} was received	INFORMATIONAL
System	System reboot occurred reason: `<reboot cause>` performed by user: `<user who performed reboot>` reboot time: `<time when reboot occurred>` For a list of reboot causes, see Possible Reboot Causes.	INFORMATIONAL
ASIC name	PSC detected failure	MAJOR

9.5.2 Clearing Events

9.5.2.1 Receiving Clearing Events

For most WARNING/CRITICAL events, the system will send a "Clearing" event once the issue is resolved. If the system experiences two or more issues for a component, it will send an event about the last issue, and once all the issues are resolved, it will send a "Clearing" event for the last issue only. Once you receive a "Clearing" event, all issues for the component are resolved.

Example:

- PSU2 is out of power
- PSU2 is missing or not available
- Cleared: PSU2 is missing or not available

9.5.2.2 Resending Unresolved Issues

If one of the initial issues for the component still exists after the last one was resolved, the system will resend the issue that still exists.

Example:

- PSU2 is out of power
- PSU2 is missing or not available

- PSU2 is out of power

9.5.2.3 Backward Compatibility

Backward compatibility is preserved, and in the case of clearing the issue for the component, the system will also send generic events. Please consider avoiding generic messages, as they will be dropped in future releases.

Example:

- Cleared: PSU2/FAN speed is out of range
- Hardware component goes back to normal

9.5.2.4 System Reboot

After a reboot, the system does not clear any pre-boot events and assumes everything is cleared.

9.5.3 Detailed Table of Events

Event Category	Event Type ID ("event" in gNMI)	Severity	Resource ("component" in gNMI)	Text	Failure Reason	Suggested Repair Flow
Fan-Related Events						
Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"FAN1/1 speed is out of range, speed=40%, range=[50,100]"	Fan speed out of range	• Collect tech-support and submit NVIDIA support ticket. • Consider number of faulty fans: more than one fan requires immediate maintenance. • Power-cycle the switch. • If persists, submit NVIDIA support ticket to replace fan module.
Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"FAN1/1 is not working"	Fan status is not okay (status in the hardware)	
Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"Failed to get actual speed data for FAN1/1"	Failed to get some information of fan data from hardware	
Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"Failed to get target speed data for FAN1/1"	Failed to get some information of fan data from hardware	
Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"Failed to get speed tolerance for FAN1/1"	Failed to get some information of fan data from hardware	

Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"Failed to get speed status for FAN1/1"	Failed to get some information of fan data from hardware	
Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"FAN1/1 is missing"	Fan is missing	
Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"FAN1/1 direction is not aligned with exhaust direction intake"	Fan direction is not aligned with other fans	
Fan failure	HEALTH_NOT_OK	WARNING	FAN1/1	"Invalid fan speed data for FAN1/1, speed=0x, target=50, tolerance=100"	Invalid speed	
Fan health	HEALTH_OK	INFORMATIONAL	FAN1/1	“HW component goes back to normal”	Fan is back to normal state	N/A
ASIC-Related Events						
ASIC failure	HEALTH_NOT_OK	WARNING	ASIC-HEALTH	Switch ASIC in fatal state	ASIC in fatal	• Collect tech-support and submit NVIDIA support ticket. • Reboot the system. • If persists, power-cycle system.
ASIC failure	SYSTEM_FATAL_DETECTED	CRITICAL	System	System fatal state detected	Detect ASIC in fatal	
ASIC failure	HEALTH_NOT_OK	WARNING	ASIC1	ASIC1 temperature is too hot, temperature=120, threshold=105	ASIC temp too high	• Collect tech-support and submit NVIDIA support ticket. • Continue to monitor switch temperature.
ASIC failure	SYSTEM_FATAL_REMEDY	MAJOR	System	Restart all syncd-ibv0 dockers	ASIC in fatal preforming reboot of dockers	N/A
ASIC failure	SYSTEM_FATAL_REMEDY	MAJOR	System	Performing reboot	ASIC in fatal preforming reboot	
ASIC health	HEALTH_OK	INFORMATIONAL	ASIC1	“HW component goes back to normal”	ASIC1 is back to normal state	

ASIC health	SYSTEM_FATAL_RECOVERED	INFORMATIONAL	System	System recovered from fatal state	Recoverd from fatal	
ASIC security irregularity	ASIC_SECURITY_IRREGULARITY	MAJOR	ASIC1	"PSC detected failure"	The system has detected an irregularity in physical monitoring	N/A
Leakage-Related Events						
Leakage	LEAKAGE	CRITICAL	LEAKAGE-1	Leakage detected, inspect for water leakage and consider power down switch tray	Detected leakage	• Collect tech-support and submit NVIDIA support ticket. • For additional instructions refer to NVONLINE 1115991 chapter "NVIDIA MGX Leak Detection Strategy and Remediation" **NOTE:** Relevant only for liquid-cooled-based systems.
Leakage	HEALTH_NOT_OK	WARNING	LEAKAGE-1	LEAKAGE-1 detected leakage	Detected leakage	
Voltage-Related Events						
Voltage	HEALTH_NOT_OK	WARNING	<Voltage-sensor-name>	Sensor voltage is out of range, voltage={}, range=[{},{}]	Voltage sensor not in range	• Collect tech-support and submit NVIDIA support ticket. • Power cycle the switch. • If persists, consider replacing the system.
Voltage	HEALTH_NOT_OK	WARNING	<Voltage-sensor-name>	Sensor status is failed	Voltage sensor status in hardware is failed	
Voltage	HEALTH_OK	INFORMATIONAL	<Voltage-sensor-name>	“HW component goes back to normal”	Voltage sensor value is back to normal state	N/A
Temperature-Related Events						

Temperature	HEALTH_NOT_OK	WARNING	<Temp-sensor-name>	<Temp-sensor-name> temperature is too hot, temperature={}, threshold={}	Temperature too hot	• Collect tech-support and submit NVIDIA support ticket. • Power cycle the switch. • If persists, see if the sensor is Ambient-MNG-Temp. If it is, check the environmental conditions (CDU and DC temperature). • If persists, consider replacing the system.
Temperature	HEALTH_NOT_OK	WARNING	<Temp-sensor-name>	Sensor status is failed	Sensor status in hardware is failed	
Temperature	HEALTH_OK	INFORMATIONAL	<Temp-sensor-name>	“HW component goes back to normal”	Temperature sensor value is back to normal state	N/A
System-Services-Related Events						
services	HEALTH_NOT_OK	WARNING	<container-name>	Container '<container-name>' is not running	Container is not running	Collect tech-support and submit NVIDIA support ticket.
services	HEALTH_OK	INFORMATIONAL	<container-name>	“Service goes back to normal”	Service goes back to normal state	N/A
services	CPU_USAGE	WARNING		CPU usage x% is above expected threshold y%	CPU usage is larger than the expected usage	• Collect techsupport and submit NVIDIA support ticket.
services	MEMORY_USAGE	WARNING		Memory usage x% is above expected threshold y%	Memory usage is larger than the expected usage	• Container will be restarted automatically. • If persists, collect techsupport and submit NVIDIA support ticket.
services	CPU_USAGE	INFORMATIONAL		CPU usage is back to normal	CPU usage is back to normal state	N/A
services	MEMORY_USAGE	INFORMATIONAL		Memory usage is back to normal	Memory usage is back to normal state	N/A
System-Initialization-Related Events						

Init flow	SYSTEM_STATE_DOWN	CRITICAL	System	System is not ready—one or more services are not up	Some services are not up as part of init	• Collect tech-support and submit NVIDIA support ticket. • If sensor is Ambient-MNG-Temp: • Check environmental conditions (CDU (if exists) and DC temperature). • If persists, power-cycle the switch. • If still persists, replace the switch.
Init flow	SYSTEM_STATE_FAILED	MAJOR	System	System is not ready—one or more services failed	Some services failed as part of init	
Init flow	SYSTEM_STATE_UP	INFORMATIONAL	System	System is ready	System finished initialization and is ready	N/A
Interface-Related Informational Events						
interface	INTERFACE_ADMIN_STATUS	INFORMATIONAL	<interface_name>	"Interface admin state is {admin_state}"	Informs of admin state change of interface	N/A
interface	INTERFACE_OPER_STATUS	INFORMATIONAL	<interface_name>	"Interface operational state is {up or down}"	Informs of operational state change of interface	N/A
interface	INTERFACE_LOGICAL_STATE	INFORMATIONAL	<interface_name>	"Interface logical state is {logical_state}"	Informs of logical state change of interface	N/A
System-Health-Related Events (The below events are summary and accompany the specific errors that were detailed above)						
system	HEALTH_SUMMARY_NOT_OK_CRITICAL	CRITICAL	System	Health status is not ok	Have some health not okay event—critical (e.g. leakage)	Collect tech-support and submit NVIDIA support ticket.
system	HEALTH_SUMMARY_NOT_OK	WARNING	System	Health status is not ok	Have some health not okay event—warning	
System	HEALTH_SUMMARY_OK	INFORMATIONAL	System	Health status is ok	System health with no issue	N/A
Transceiver-Related Events						

Transceiver failure	HEALTH_NOT_OK	WARNING	sw1	"Transceiver's temperature is higher than critical threshold [actual = 100, high threshold = 80]"	Temperature is critically high	N/A
Transceiver failure	HEALTH_NOT_OK	WARNING	sw1	"Transceiver's temperature is lower than critical threshold [actual = 1, low threshold = 5]"	Temperature is critically low	N/A
Transceiver health	HEALTH_OK	INFORMATIONAL	sw1	"HW component goes back to normal"	Transceiver's temperature is good now	N/A

Event Category	Event Type ID ("event" in gNMI)	Severity	Resource ("component" in gNMI)	Text	Failure Reason	Suggested Repair Flow
Transceiver-Related Events						
Transceiver failure	HEALTH_NOT_OK	WARNING	sw1	"Unsupported cable detected"	Inserted cable vendor is not recognized	N/A
Transceiver failure	HEALTH_NOT_OK	WARNING	els1	"HW Component health is not ok: Laser failure"	ELS laser isn't functioning properly	Replace ELS

Event Category	Event Type ID ("event" in gNMI)	Severity	Resource ("component" in gNMI)	Text	Failure Reason	Suggested Repair Flow
Power-Supply-Unit-Related Events						
PSU	HEALTH_NOT_OK	WARNING	<PSU-name>	<PSU-name> is missing—Unpopulated PSU slot	PSU expected to be in the system, but PSU slot is empty	Insert PSU/dummy PSU to the unpopulated PSU slot
PSU	HEALTH_NOT_OK	WARNING	<PSU-name>	<PSU-name> is out of power	PSU is out of power	

Event Category	Event Type ID ("event" in gNMI)	Severity	Resource ("component" in gNMI)	Text	Failure Reason	Suggested Repair Flow
PSU	HEALTH_NOT_OK	WARNING	<PSU-name>	<PSU-name> temperature is too hot, temperature={}, threshold={}	Temperature too hot	
PSU	HEALTH_NOT_OK	WARNING	<PSU-name>	<PSU-name> voltage is out of range, voltage={}, range=[{},{}]	Voltage is out of range	
PSU	HEALTH_NOT_OK	WARNING	<PSU-name>	<PSU-name> System power exceeds threshold ({}w)	Power exceeds threshold	
PSU	HEALTH_NOT_OK	WARNING	<PSU-name>	<PSU-name> Power supply is not providing power	No power from PSU	
PSU	HEALTH_OK	INFORMATIONAL	<PSU-name>	HW component goes back to normal	Health issue was resolved	N/A

Event Category	Event Type ID ("event" in gNMI)	Severity	Resource ("component" in gNMI)	Text	Failure Reason	Suggested Repair Flow
PS-Redundancy-Related Events						
Power redundancy policy	HEALTH_NOT_OK	WARNING	N/A	Power redundancy policy no-redundancy requires at least <minimal-number-of-PSUs-per-system> working power supplies, currently system has only <number-of-working-PSUs>	Insufficient number of PSUs relative to the current no-redundancy policy	Insert at least the minimal amount of PSUs required. Minimal PSU amount can be found using the following command: nv show platform ps-redundancy

Event Category	Event Type ID ("event" in gNMI)	Severity	Resource ("component" in gNMI)	Text	Failure Reason	Suggested Repair Flow
Power redundancy policy	HEALTH_NOT_OK	WARNING	N/A	Power redundancy policy ps-redundant requires at least <minimal-number-of-PSUs-per-system + 1> working power supplies, currently system has only <number-of-working-PSUs>	Insufficient number of PSUs relative to the current ps-redundant policy	Insert at least one more PSU than the minimal amount of PSUs required. Minimal PSU amount can be found using the following command: nv show platform ps-redundancy
Power redundancy policy	HEALTH_NOT_OK	WARNING	N/A	Power redundancy policy grid-redundant requires all <number-of-PSUs-in-the-system> power supplies to be working, currently system has only <number-of-working-PSUs>	Insufficient number of PSUs relative to the current grid-redundant policy	Insert PSUs to all PSU slots in the system.
Power redundancy policy	HEALTH_OK	INFORMATIONAL	N/A	HW component goes back to normal	Health issue was resolved	N/A

9.5.4 Event Management Commands

- Event Management Commands

9.5.5 Event Management Commands

9.5.5.1 nv show system events

	nv show system events Display events generated by the system.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.5002 Added Etc/UTC to event timestamp in output
Example	admin@nvos:~$nv show system events operational table-occupancy 80 table-size 1000 Events Event ID Severity Component Description Timestamp 80 INFORMATIONAL System Health status is ok 2024-04-23 08:29:05 Etc/UTC 79 INFORMATIONAL PSU2/FAN HW component goes back to normal 2024-04-23 08:29:05 Etc/UTC 78 INFORMATIONAL PSU1/FAN HW component goes back to normal 2024-04-23 08:29:05 Etc/UTC 77 WARNING PSU1/FAN PSU1/FAN speed is out of range, speed=30%, range=[35,100] 2024-04-23 08:29:02 Etc/UTC 76 WARNING PSU2/FAN PSU2/FAN speed is out of range, speed=30%, range=[35,100] 2024-04-23 08:28:56 Etc/UTC 75 WARNING PSU2/FAN PSU2/FAN speed is out of range, speed=29%, range=[35,100] 2024-04-23 08:28:50 Etc/UTC 74 WARNING PSU1/FAN PSU1/FAN speed is out of range, speed=29%, range=[35,100] 2024-04-23 08:28:50 Etc/UTC
REST API	GET https://<ip>/nvue_v1/system/events
Related Commands	nv action clear system events
Notes	It shows only the last 50 events in order to not flood the screen with multiple lines at the same time. To get more events, 'last' option can be used.

9.5.5.2 nv show system events --last

	nv show system events --last [number] Show requested last events.	
Syntax Description	number	Requested number of events to show
Default	20 events	
History	25.02.2002 25.02.5002 Updated command syntax. JSON no longer has "last" as a key.	

Example	admin@nvos:~$ nv show system events --last Event ID Severity Component Description Timestamp -------- ------------ ---------- ---------- ---------- 63529 WARNING swss-ibv00 Container 'swss-ibv00' is not running 2025-05-16 08:05:44 63528 WARNING syncd-ibv00 Container 'syncd-ibv00' is not running 2025-05-16 08:05:41 63527 WARNING pmon Container 'pmon' is not running 2025-05-16 08:05:41 63526 INFORMATIONAL pmon Service goes back to normal 2025-05-16 08:05:26 63525 INFORMATIONAL pmon Cleared: Container 'pmon' is not running 2025-05-16 08:05:26
REST API	GET https://<ip>/nvue_v1/system/events?last/{number}
Related commands	nv action clear system events
Notes	If no number is specified, this command will show the last 20 entries from the table.

9.5.5.3 nv show system events --recent

	nv show system events --recent [minutes] Show events in the last requested minutes.	
Syntax Description	minutes	Time in past minutes to show events from
Default	5 minutes	
History	25.02.2002 25.02.5002 Updated command syntax. JSON no longer has "recent" as a key.	
Example	admin@nvos:~$ nv show system events --recent 10 Event ID Severity Component Description Timestamp -------- ------------ ---------- ---------- ---------- 63529 WARNING swss-ibv00 Container 'swss-ibv00' is not running 2025-05-16 08:05:44	
REST API	GET https://<ip>/nvue_v1/system/events?recent/{minutes}	
Related commands	nv action clear system events	
Notes	If no minutes are specified, this command will display events from the past 5 minutes.	

9.5.5.4 nv set/unset system events table-size

	nv set/unset system events table-size <number-of-lines> Set/unset events table size.	
Syntax Description	number-of-lines	Number of lines shown in the events table
Default	1000 lines	
History	25.02.2002	

Example	 admin@nvos:~$nv set system events table-size 5000 admin@nvos:~$nv unset system events table-size
REST API	PATCH https://<ip>/nvue_v1/system/events/table-size/{number-of-lines}
Related commands	nv show system events
Notes	

9.5.5.5 nv action clear system events

	nv action clear system events Clear events.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	 admin@nvos:~$nv action clear system events Action executing ... Event table has been cleared Action succeeded	
REST API	POST https://<ip>/nvue_v1/system/events	
Related Commands	nv show system events	
Notes		

9.6 Statistics

NVOS collects samples and saves system statistics data. Sampling data is collected for tech-support.

Supporting CLI:

- Show, upload, clear data collected.
- Configure sampling configuration.
    - Enable/disable sampling
    - Sampling interval
    - How long to save samples (cyclic buffer)

Statistics are split into multiple categories:

Category	Statistic	Description
Temperature	asic-<x>	Asic temperature (per asic)
	ambient COMEX	Ambient comex temperature
	Ambient Fan Side	Ambient fan side temperature
	Ambient Port Side	Ambient port-side temperature
	CPU Core <x>	CPU core temperature (per cpu)

Category	Statistic	Description
	CPU Pack	CPU pack temperature
	PSU-<x>	PSU temperature (per PSU)
	Module <idx>	Module temperatureerature
CPU	Free-ram	Free system ram memory
	CPU utilization	CPU utilization over the last 5 minutes (can be either 1/5/15)
	Reboot count	Number of reboots
Disk	Free space	Free space under "/"
	Wear level	Disk remaining time left
	Write rate	Average disk write/sec
	Read rate	Average disk read/sec
	Program fail count	Counts the number of flash program failures
	Erase fail count	counts the number of failed data deletion attempts
	Uncorrectable error count	count of errors that are impossible to recover
Power	Psu-power <x>	PSU power [W]
	Psu-current <x>	PSU current [A]
Fan	Fan-speed <x>	Fan speed [%]
Mgmt-interface	Eth<x> tx bytes	TX bytes diff from last sample
	Eth<x> rx bytes	RX bytes diff from last sample

9.6.1 Statistics Commands

- Statistics Commands

9.6.2 Statistics Commands

9.6.2.1 nv show system stats

	nv show system stats Display configuration for system statistics collection reports.
Syntax Description	N/A
Default	N/A
History	25.02.2002 25.02.3000 Updated command output
Example	admin@nvos:~$ nv show system stats operational applied pending ---------- ---------- --------------- --------------- state enabled enabled enabled [category] cpu cpu [category] disk disk [category] fan fan [category] mgmt-interface mgmt-interface [category] power power [category] temperature temperature [category] voltage voltage
REST API	GET https://<ip>/nvue_v1/system/stats
Related Commands	nv set/unset system stats state
Notes	

9.6.2.2 nv show system stats category

	nv show system stats category [<category-id>] Get statistic categories.	
Syntax Description	category-id	Display configuration for specified category.
Default	N/A	
History	25.02.2002 25.02.3000 Updated command output	
Example	admin@nvos:~$ nv show system stats category History Duration (days) Interval (minutes) State -------------- ---------------------- ------------------ ------- cpu 365 5 enabled disk 365 30 enabled fan 365 5 enabled mgmt-interface 365 5 enabled power 365 5 enabled temperature 365 5 enabled voltage 365 5 enabled admin@nvos:~$ nv show system stats category fan operational applied ---------------- ---------- ------- state enabled enabled interval 5 5 history-duration 365 365	

REST API	GET https://<ip>/nvue_v1/system/stats/category GET https://<ip>/nvue_v1/system/stats/category/< category-id >
Related Commands	nv set/unset system stats category
Notes	

9.6.2.3 nv show system stats files

	nv show system stats files [<file-name>] Display statistic report files.	
Syntax Description	file-name	Display content of specified CSV file.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show system stats files Statistic report file File path --- --- stats_fan_x_20250223_134744.csv /host/stats/stats_fan_x_20250223_134744.csv admin@nvos:~$ nv show system stats files stats_fan_x_20250223_134744.csv	
REST API	GET https://<ip>/nvue_v1/system/stats/files GET https://<ip>/nvue_v1/system/stats/files/< file-name >	
Related Commands	nv action generate system stats category	
Notes	Autocomplete returns also generated tar file (in case if it was generated before), but it can't be displayed, only *.csv files can be displayed	

9.6.2.4 nv set/unset system stats state

	nv set/unset system stats state [{ enabled | disabled }] Set/unset state.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system stats state disabled admin@nvos:~$ nv unset system stats state	
REST API	PATCH https://<ip>/nvue_v1/system/stats	
Related Commands	nv show system stats	
Notes		

9.6.2.5 nv set/unset system stats category state

	nv set/unset system stats category <category-id> state [{ enabled | disabled }] Set/unset category state.	
Syntax Description	category-id	Change state for specified category.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set system stats category fan state disabled admin@nvos:~$ nv unset system stats category fan state	
REST API	PATCH https://<ip>/nvue_v1/system/stats/category/<category-id>	
Related Commands	nv show system stats category	
Notes		

9.6.2.6 nv set/unset system stats category interval

	nv set/unset system stats category <category-id> interval Set interval in minutes.	
Syntax Description	category-id	Change sampling interval for specified category.
Default	5 (30 for disk category)	
History	25.02.2002	
Example	admin@nvos:~$ nv set system stats category fan interval 60 admin@nvos:~$ nv unset system stats category fan interval	
REST API	PATCH https://<ip>/nvue_v1/system/stats/category/<category-id>	
Related Commands	nv show system stats category	
Notes	Sampling interval is in minutes.	

9.6.2.7 nv set/unset system stats category history-duration

	nv set/unset system stats category <category-id> history-duration[{ 1–365 }] Sampling history duration in days.	
Syntax Description	category-id	Change history-duration for specified category.
	history-duration	1–365 days
Default	365	
History	25.02.2002	

Example	admin@nvos:~$ nv set system stats category fan history-duration 31 admin@nvos:~$ nv unset system stats category fan history-duration
REST API	PATCH https://<ip>/nvue_v1/system/stats/category/<category-id>
Related Commands	nv show system stats category
Notes	History duration is in days.

9.6.2.8 nv action clear system stats category

	nv action clear system stats category <category-id> Clear statistic category sampled data.	
Syntax Description	category-id	Category to be cleared.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action clear system stats category fan	
REST API	POST https://<ip>/nvue_v1/system/stats/category/<category-id>	
Related Commands	nv action generate system stats category	
Notes		

9.6.2.9 nv action clear system stats category

	nv action clear system stats category <category-id> Clear statistic category sampled data.	
Syntax Description	category-id	Category to be cleared.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action clear system stats category fan	
REST API	POST https://<ip>/nvue_v1/system/stats/category/<category-id>	
Related Commands	nv action generate system stats category	
Notes		

9.6.2.10 nv action generate system stats category

	nv action generate system stats category <category-id> Generate statistic report file.

Syntax Description	category-id	Category to be generated CSV file for.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action clear system stats category fan	
REST API	POST https://<ip>/nvue_v1/system/stats/category/<category-id>	
Related Commands	nv show system stats files	
Notes		

9.6.2.11 nv action generate system stats

	nv action generate system stats Generate tar file with reports from all categories.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action generate system stats	
REST API	POST https://<ip>/nvue_v1/system/stats	
Related Commands	nv show system stats files	
Notes		

9.6.2.12 nv action clear system stats

	nv action clear system stats Clear statistic sampled data for all categories.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action clear system stats	
REST API	POST https://<ip>/nvue_v1/system/stat	
Related Commands	nv action generate system stats	
Notes		

9.6.2.13 nv action delete system stats files

	nv action delete system stats files <file-name> Delete available stats report file.	
Syntax Description	file-name	Report file to be deleted.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action delete system stats files stats_fan_x_20250223_134744.csv	
REST API	POST https://<ip>/nvue_v1/system/stats/files/<file-name>	
Related Commands	nv action generate system stats category nv show system stats files	
Notes		

9.6.2.14 nv action upload system stats files

	nv action upload system stats files <file-name> <remote-url> Upload available stats report files to remote location.	
Syntax Description	file-name	Report file name to be uploaded.
	remote-url	Destination image file name Remote url path to upload a file to. Format: [protocol]://username[:password]@hostname/path/filename Supported protocols: SCP, FTP, SFTP, and HTTPS
Default	N/A	
History	25.02.2002 25.02.4002 Added HTTPS support in remote-url	
Example	admin@nvos:~$ nv action upload system stats files stats_fan_x_20250223_134744.csv scp://user@host/path/to/folder	
REST API	POST https://<ip>/nvue_v1/system/stats/files/<file-name>	
Related Commands	nv show system stats files	
Notes		

9.7 Technical Support

NVOS supports generating technical support files to assist with troubleshooting and issue resolution. These files contain critical system logs and diagnostic information that help support teams analyze and address reported issues efficiently.

It is recommended that users generate a technical support file whenever they encounter an issue and include it when filing a support request.

9.7.1 How to Generate and Upload a Tech-Support File

For troubleshooting your transceiver, generating and uploading a tech-support file is essential. This file includes detailed diagnostics to aid technical support in resolving your issues.

1. Generate the file.

```
admin@nvos:~$ nv action generate system tech-support
Generating system tech-support file, it might take a few minutes...
Generated tech-support /var/dump/nvos_dump_nvos_20220601_114011.tar.gz
Action succeeded
```

2. Upload generated file to a designated location.

```
admin@nvos:~$ nv action upload system tech-support files nvos_dump_nvos-switch_20230120_151401.tar.gz
scp://username@host-name/home/logs/
Password:
Action executing ...
Upload file nvos_dump_nvos-switch_20230120_151401.tar.gz
Action executing ...
File upload successfully
Action succeeded
```

9.7.2 Technical Support Commands

- Technical Support Commands

9.7.3 Technical Support Commands

9.7.3.1 nv show system tech-support files

	nv show system tech-support files Show the created tech-support files on the switch.	
Syntax Description	N/A	
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv show system tech-support files File name Tech-support file path --- nvos_dump_nvos-switch_20230120_151401.tar.gz /host/dump/nvos_dump_nvos-switch_20230120_151401.tar.gz	
REST API	GET https://<ip>/nvue_v1/system/tech-support/files	

Related Commands	nv action generate system tech-support nv action delete system tech-support files
Notes	

9.7.3.2 nv action generate system tech-support

	nv action generate system tech-support {--since <time>} Action to generate a tech-support file on switch.	
Syntax Description	--since	Collect logs and cores only from the given since date
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action generate system tech-support Generating system tech-support file, it might take a few minutes... Generated tech-support /var/dump/nvos_dump_jaguar-70_20220601_114011.tar.gz Action succeeded admin@nvos:~$ nv action generate system tech-support --since=1/1/2022 Generating system tech-support file, it might take a few minutes... Generated tech-support /var/dump/nvos_dump_jaguar-70_20220601_114315.tar.gz Action succeeded	
REST API	POST https://<ip>/nvue_v1/system/tech-support	
Related Commands	nv show system tech-support files nv action delete system tech-support files nv action upload system tech-support files	
Notes		

9.7.3.3 nv action delete system tech-support files

	nv action delete system tech-support files <file-name> Delete tech-support file from the file system.	
Syntax Description	file-name	Name of the tech-support file.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action delete system tech-support files nvos_dump_nvos-switch_20230120_151401.tar.gz File delete successfully Action succeeded	
REST API	POST https://<ip>/nvue_v1/system/tech-support/files/{file-name}	
Related Commands	nv action show system tech-support files nv action generate system tech-support nv action upload system tech-support files	
Notes		

9.7.3.4 nv action upload system tech-support files

	nv action upload system tech-support files <file-name> <remote-url> Upload tech-support file to the remote server.	
Syntax Description	file-name	Name of the tech-support file.
	remote-url	Destination image file name Remote url path to upload a file to. Format: [protocol]://username[:password]@hostname/path/filename Supported protocols: SCP, FTP, SFTP, and HTTPS
Default	N/A	
History	25.02.2002 25.02.4002 Added HTTPS support in remote-url	
Example	admin@nvos:~$ nv action upload system tech-support files nvos_dump_nvos-switch_20230120_151401.tar.gz scp://username@host-name/home/logs/ Password: Upload file 1 File upload successfully Action succeeded	
REST API	POST https://<ip>/nvue_v1/system/tech-support/files/{file-name}	
Related Commands	nv action show system tech-support files nv action generate system tech-support nv action delete system tech-support files	
Notes		

9.8 Troubleshooting

9.8.1 Resetting NVOS Password

To reset forgotten password of default user accounts, see Reset Local Users' Passwords section.

9.8.2 Image Upgrade Recovery

If the system encounters issues after an image upgrade, the user can switch back to the old partition.

1. Check the current partition.

```
admin@nvos:~$ nv show system image
```

```
    operational
---------- ------------------------
current  nvos-25.02.1500
next   nvos-25.02.1500
partition1 nvos-25.02.1500
partition2 nvos-25.02.1400
```

2. Change to the other partition.

```
admin@nvos:~$ nv action boot-next system image partition2
admin@nvos:~$ nv show system image

    operational
---------- ------------------------
current  nvos-25.02.1500
next   nvos-25.02.1400
partition1 nvos-25.02.1500
partition2 nvos-25.02.1400
```

3. Reboot the system.

```
admin@nvos:~$ nv action reboot system
```

9.8.3 Management Interface Has No IPv4/6

Check that DHCP is running on the relevant protocol:

```
admin@nvos:~$ nv show interface eth0 ip
                operational                             applied
--------------  --------------------------------------  -------
vrf             default                                 default
arp-timeout     1800                                    1800
autoconf        enabled                                 enabled
dhcp-client
  state         enabled                                 enabled
  set-hostname  enabled                                 enabled
  is-running    yes
  has-lease     no
dhcp-client6
  state         enabled                                 enabled
  set-hostname  enabled                                 enabled
  is-running    yes
  has-lease     yes
[address]       10:10:10:10:10:10:10:10/10
[address]       fdfd:fdfd:7:80:eaeb:d3ff:fe4b:70b8/64
[gateway]
```

If the interface does not have a lease, run the following action to renew the DHCP lease:

```
admin@nvos:~$ nv action renew interface eth0 ip dhcp-client
Action executing ...
Renewing DHCP lease for eth0, connection may be interrupted
Action executing ...
DHCP lease for eth0 was renewed
Action succeeded
```

9.8.4 System Fatal Recovery

The system has mechanism to detect if ASIC encountered health/firmware burn issue and try to recover from it.

During the fatal detection and recovery, events will be raised as well. For more information, see ASIC-Related Events in the Event Managment section.

9.8.4.1 Detecting a Fatal State

The system's fatal state is indicated in the CLI prompt and in the nv show system health command.
Example:

```
[System_Fatal_State]admin@nvos~$ nv show system health
            operational  applied
----------  -----------  -------
status      FATAL
status-led  amber


Health issues
================
    Component    Status information
    -----------  --------------------------
    ASIC-HEALTH  Switch ASIC in fatal state.
```

9.8.4.2 Automatic Recovery Mechanism

The system has an internal mechanism to recover from a fatal state without user intervention. The recovery process involves the following steps:

1. Restart the ASICs of the system.
2. If restarting the ASICs does not resolve the issue, the system will attempt to recover through a system reboot.
   If after the reboot system still encounters ASIC issues, another reboot will be performed.
3. After the second reboot, the system will start without configuring the ASICs, leaving all ports down. NVOS is running, so logs can be collected for analysis.
4. To try to revive the switch, perform a system reboot by running nv action reboot system.
5. If system entered fatal state again, please contact NVIDIA's support team.

Any reboot initiated by the user will also reset the system's fatal detection and recovery mechanism. This process starts the recovery steps from the beginning.

9.8.4.3 Recovery Timeframe

- After the recovery steps are completed and the system remains operational for 10 minutes without any health issues, it will exit the fatal state.
- During this 10-minute observation period, the system may still appear in a fatal state as reflected in the CLI prompt and system health command.
- Once the system exits the fatal state, the CLI prompt and system health command will confirm the recovery.

10 InfiniBand Switching

The following pages provide information on configuring InfiniBand protocols and features.

10.1 InfiniBand Interface

10.1.1 Transceiver Information

NVOS offers the option of viewing the transceiver information of a module or cable connected to a specific interface. The information is a set of read-only parameters burned onto the EEPROM of the transceiver by the manufacturer. The parameters include identifier (connector type), cable type, and additional inventory attributes. For optical transceivers, there are also diagnostic parameters such as temperature, voltage, and information about channels.

10.1.1.1 Configure Connection-Mode In an InfiniBand Interface

The connection-mode attribute indicates to which peer this port is connected. This feature is dedicated to Q3200-RA system which can be connected to NDR systems (or older) and to XDR systems.
In case the port is connected to XDR systems, the connection-mode should be XDR. If it connected to NDR system, the connection-mode should be NDR. By default, all the ports are configured to connection-mode XDR.
When connecting port to NDR system, the port connection-mode needs to be changed with the set command (nv set interface link connection-mode) to 'ndr' in order to have link up with full bandwidth. Changing port connection-mode will move the port to down and up again when configuration ends.

10.1.2 InfiniBand Interface Commands

10.1.3 InfiniBand Interface Commands

10.1.3.1 nv show interface

	nv show interface [interface-id] Displays details of a single InfiniBand interface. When no interace is specified, the output shows all interfaces.	
Syntax Description	interface-id	Name of the InfiniBand interface to display
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show interface sw9p1
	                    operational applied
	-------------------- ------------------- --------
	link
	  auto-negotiate on on
	  duplex full
	  speed auto
	  counters
	    in-bytes 0 Bytes
	    in-pkts 0
	    in-drops 0
	    in-errors 0
	    out-bytes 0 Bytes
	    out-pkts 0
	    out-drops 0
	    out-errors 0
	    in-symbol-errors 0
	    out-wait 0
	  [diagnostics]
	  mtu 4096
	  op-vls VL0-VL7
	  lanes 1X,2X,4X
	  ib-speed auto
	  state down up
	  supported-ib-speed sdr,fdr,edr,hdr,ndr
	  max-supported-mtu 4096
	  physical-state Polling
	  logical-state Down
	  supported-lanes 1X,2X,4X
	  vl-capabilities VL0-VL7
	  ib-subnet infiniband-default
	  round-trip-latency 248 ns
	type ib ib</pre>
	<pre>admin@nvos:~$ nv show interface
	Interface State Speed MTU Type Description IB Speed IB Subnet
	Logical State Physical State Summary
	--------- ----- ----- ----- -------- ----------- -------- ------------------
	------------- -------------- --
	eth0 up 1G 1500 eth
	IPv4 Address: 192.168.10.45/24
	
	IPv6 Address: fd00:abcd:1:22::2000:af12/128
	
	IPv6 Address: fd00:abcd:1:22:d812:cegg:feba:1234/64
	eth1 up 1G 1500 eth
	IPv4 Address: 192.168.10.46/24
	
	IPv6 Address: fd00:abcd:1:22::2000:bf99/128
	
	IPv6 Address: fd00:abcd:1:22:d812:cegg:feba:1235/64
	fnm1 up 800G 4096 fnm xdr
	infiniband-default Active LinkUp
	ib0 up 10G 2044 ipoib
	lo up 65536 loopback
	IPv4 Address: 127.0.0.1/16
	
	IPv6 Address: ::1/128
	sw1p1 up 800G 4096 ib xdr
	infiniband-default Active LinkUp
	sw1p2 up 800G 4096 ib xdr
	infiniband-default Active LinkUp
	sw2p1 up 800G 4096 ib xdr
	infiniband-default Active LinkUp
	sw2p2 up 800G 4096 ib xdr
	infiniband-default Active LinkUp
	sw3p1 down ib infiniband-default Down
	Disabled
	sw3p2 down ib infiniband-default Down
	Disabled
	sw4p1 down ib infiniband-default Down
	Disabled
	sw4p2 down ib infiniband-default Down
	Disabled
	sw5p1 down ib infiniband-default Down
	Disabled
	sw5p2 down ib infiniband-default Down
	Disabled
	sw6p1 down ib infiniband-default Down
	Disabled</pre>

REST API	GET https://<id>/nvue_v1/interface/{interface-id}
Related Commands	set interface show interface management
Notes	The data presented here is for an InfiniBand interface. If the id of a different type of interface is provided, the output will be different.

10.1.3.2 nv show interface link

	nv show interface <interface-id> link {state | counters | phy-diag | phy-detail} Displays link information of a single InfiniBand interface.	
Syntax Description	interface-id	Name of the InfiniBand interface to display.
	state	Show only the data relating to state.
	counters	Show only the data relating to counters.
	phy-diag	Show PHY diagnostics and FSM states, part of AMBER
	phy-detail	Show PHY statistics and BER measurements, part of AMBER
Default	N/A	
History	25.02.2002	

Example

```
admin@nvos:~$ nv show interface sw23p2 link
                    operational          applied
------------------  -------------------  --------
auto-negotiate      on                   on
duplex                                   full
speed               400G                 auto
counters
  in-bytes          5.06 KB
  in-pkts           18
  in-drops          0
  in-errors         0
  out-bytes         5.06 KB
  out-pkts          18
  out-drops         0
  out-errors        0
  in-symbol-errors  0
  out-wait          0
[diagnostics]
mtu                 4096                 4096
op-vls              VL0-VL3              VL0-VL7
lanes               4X                   1X,2X,4X
ib-speed            ndr                  auto
state               up                   up
supported-ib-speed  sdr,fdr,edr,hdr,ndr
max-supported-mtu   4096
physical-state      LinkUp
logical-state       Active
supported-lanes     1X,2X,4X
vl-capabilities     VL0-VL7
ib-subnet           infiniband-default
round-trip-latency  312 ns
```

```
admin@nvos:~$ nv show interface sw23p2 link state
operational  applied
-----------  -------
up           up

admin@nvos:~$ nv show interface sw1p1 link counters
                  operational
----------------  -----------
in-bytes          0 B
in-drops          0
in-errors         0
in-pkts           0
out-bytes         0 B
out-drops         0
out-errors        0
out-pkts          0
in-symbol-errors  0
out-wait          0
```

```
admin@nvos:~$ nv show interface sw1p1 link phy-diag
                                      operational
------------------------------------  -----------
pd-fsm-state                          0
eth-an-fsm-state                      0
phy-hst-fsm-state                     LINKUP
psi-fsm-state                         IDLE
phy-manager-link-width-enabled        256
phy-manager-link-proto-enabled        1:20000
core-to-phy-link-width-enabled        256
core-to-phy-link-proto-enabled        1:800000
cable-proto-cap-ext                   1:0
loopback-mode                         NONE
retran-mode-request                   0
retran-mode-active                    0
fec-mode-request                      2048
profile-fec-in-use                    255
phy-manager-state                     ACTIVE
sync-header-error-counter             0
port-local-physical-errors            0
port-malformed-packet-errors          0
port-buffer-overrun-errors            0
port-dlid-mapping-errors              0
port-vl-mapping-errors                0
port-looping-errors                   0
port-inactive-discards                0
port-neighbor-mtu-discards            0
plr-rcv-codes                         73634260371
plr-rcv-codes-err                     0
plr-rcv-uncorrectable-code            0
plr-xmit-codes                        73634123139
plr-xmit-retry-codes                  0
plr-xmit-retry-events                 0
plr-sync-events                       0
plr-codes-loss                        0
plr-xmit-retry-events-within-t-sec-max  0
plr-bw-loss-percent                   0.0
rq-general-error                      0
ib-phy-fsm-state                      DISABLED
zero-hist                             5
successful-recovery-events            0

admin@nvos:~$ nv show interface sw1p1 link phy-detail
                           operational
```

```
-------------------------  ---------------
time-since-last-clear-min  1945856
phy-received-bits          778342400000000
symbol-errors              0
effective-errors           0
phy-raw-errors-lane0       1906686
phy-raw-errors-lane1       1659554
phy-raw-errors-lane2       0
phy-raw-errors-lane3       0
phy-raw-errors-lane4       0
phy-raw-errors-lane5       0
phy-raw-errors-lane6       0
phy-raw-errors-lane7       0
raw-ber                    4E-9
symbol-ber                 15E-255
effective-ber              15E-255
raw-ber-lane0              4E-9
raw-ber-lane1              4E-9
raw-ber-lane2              0E-0
raw-ber-lane3              0E-0
raw-ber-lane4              0E-0
raw-ber-lane5              0E-0
raw-ber-lane6              0E-0
raw-ber-lane7              0E-0
rs-num-corr-err-bin0       151471657460
rs-num-corr-err-bin1       1690966
rs-num-corr-err-bin2       937603
rs-num-corr-err-bin3       16
rs-num-corr-err-bin4       5
rs-num-corr-err-bin5       0
rs-num-corr-err-bin6       0
rs-num-corr-err-bin7       0
rs-num-corr-err-bin8       0
rs-num-corr-err-bin9       0
rs-num-corr-err-bin10      0
rs-num-corr-err-bin11      0
rs-num-corr-err-bin12      0
rs-num-corr-err-bin13      0
rs-num-corr-err-bin14      0
rs-num-corr-err-bin15      0
```

REST API	GET https://<id>/nvue_v1/interface/{interface-id}/link GET https://<id>/nvue_v1/interface/{interface-id}/link/counters GET https://<id>/nvue_v1/interface/{interface-id}/link/state GET https://<ip>/nvue_v1/interface/{interface-id}/link/phy-diag GET https://<ip>/nvue_v1/interface/{interface-id}/link/phy-detail
Related Commands	set interface link
Notes	

10.1.3.3 nv set interface link state

	nv set interface <interface-id> link state {value} nv unset interface <interface-id> link state {value} Sets the administrative link state of a given InfiniBand interface. The unset sets the administrative link state of a given InfiniBand interface to the default value of "up".	
Syntax Description	interface-id	Name of the interface whose link state to set.
	Value	New value for the link state: {up, down}
Default	up	
History	25.02.2002	
Example		
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/link/state	
Related Commands	show interface nv unset interface link	
Notes		

```
admin@nvos:~$ nv set interface sw9p1 link state down
admin@nvos:~$ nv unset interface sw9p1 link state
```

10.1.3.4 nv set interface link mtu

	nv set interface <interface-id> link mtu {value} nv unset interface <interface-id> link mtu Sets the mtu of a given InfiniBand interface. The unset form of the command sets the link mtu of a given InfiniBand interface to the default value of 4096.	
Syntax Description	interface-id	Name of the interface whose link mtu to set
	value	New value for the MTU: 256, 512 ,1024, 2048, 4096
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface sw1p1 link mtu 512 admin@nvos:~$ nv unet interface sw1p1 link mtu	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/link	
Related Commands	set interface link show interface infiniband	
Notes		

10.1.3.5 nv set interface link op-vls

	nv set interface <interface-id> link op-vls {value} nv unset interface <interface-id> link op-vls Sets the op-vls of a given InfiniBand interface. The unset for of the command sets the link op-vls of a given InfiniBand interface to the default value of VL0-VL7.	
Syntax Description	interface-id	Name of the interface whose link op-vls to set
	value	New value for the op-vls: VL0, VL0-VL1, VL0-VL3, VL0-VL7
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface sw1p1 link op-vls VL0-VL3 admin@nvos:~$ nv unet interface sw1p1 link op-vls	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/link	
Related Commands	set interface link show interface infiniband	
Notes		

10.1.3.6 nv set interface link lanes

	nv set interface <interface-id> link lanes {value} nv unset interface <interface-id> link lanes Sets the lanes of a given InfiniBand interface. The unset form of the command sets the link lanes of a given InfiniBand interface to the default value of 1X,4X.	
Syntax Description	interface-id	Name of the interface whose link lanes to set
	value	New value for the lanes: • 1X • 1X,2X • 1X,4X • 1X,2X,4X
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface sw1p1 link lanes 1X,4X admin@nvos:~$ nv unet interface sw1p1 link lanes	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/link	
Related Commands	set interface link show interface infiniband	
Notes		

10.1.3.7 nv set interface link ib-speed

	nv set interface <interface-id> link ib-speed {value} nv unset interface <interface-id> link ib-speed Sets the ib-speed of a given InfiniBand interface. The unset form of the command sets the link ib-speed of a given InfiniBand interface to the max supported ib-speed.	
Syntax Description	interface-id	Name of the interface whose link ib-speed to set.
	value	New value for the ib-speed: • sdr—10.0Gb/s rate on 4 lane width • qdr—40.0Gb/s rate on 4 lane width • fdr—56.0Gb/s rate on 4 lane width • edr—100.0Gb/s rate on 4 lane width • hdr—200.0 Gb/s rate on 4 lane width • ndr—400.0 Gb/s rate on 4 lane width • auto—The max supported speed of the interface
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface sw1p1 link ib-speed hdr admin@nvos:~$ nv unset interface sw1p1 link ib-speed	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/link	
Related Commands	set interface link show interface infiniband	

Notes	

10.1.3.8 nv set interface link auto-negotiate

	nv set interface <interface-id> link auto-negotiate {value} nv unset interface <interface-id> link auto-negotiate Sets the auto-negotiate of a given InfiniBand interface. The unset form of the command sets the link auto-negotiate of a given InfiniBand interface to the default value which is on.	
Syntax Description	interface-id	Name of the interface whose link auto-negotiate to set.
	value	New value for the auto-negotiate: {on, off}
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface sw1p1 link auto-negotiate on admin@nvos:~$ nv unet interface sw1p1 link auto-negotiate	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/link	
Related Commands	set interface link show interface infiniband	
Notes		

10.1.3.9 nv set interface description

	nv set interface <interface-id> description {value} nv unset interface <interface-id> description {value} Sets the description of a given InfiniBand interface. The unset sets the description of a given InfiniBand interface to empty.	
Syntax Description	interface-id	Name of the interface whose description to set.
	Value	New value for the description.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface sw9p1 description "sw9p1 description" admin@nvos:~$ nv unset interface sw9p1 description admin@nvos:~$ nv unset interface sw1p1 admin@nvos:~$ nv unset interface	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}	
Related Commands	nv show interface	
Notes		

10.1.3.10 nv action clear interface link counters

	nv action clear interface <interface-id> link counters Clears the interface counters for the user running the command.	
Syntax Description	interface-id	Name of the interface whose link stats to clear. or range of interfaces (e.g. sw1-2p1-2 → sw1p1,sw1p2,sw2p1,sw2p2)
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action clear interface sw1p1 link counters	
REST API	POST https://<id>/nvue_v1/interface/{interface-id}/link/counters	
Related Commands	nv show interface <id> link counters	
Notes		

10.1.3.11 nv action clear interface counters

	nv action clear interface counters Clears all InfiniBand interfaces counters for the user running the command.
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv action clear interface counters
REST API	POST https://<id>/nvue_v1/interface
Related Commands	
Notes	

10.2 IPoIB

10.2.1 IPoIB Commands

10.2.2 IPoIB Commands

10.2.2.1 nv show interface ib0

	nv show interface {interface-id} Displays details of an IPoIB interface.	
Syntax description	interface-id	Name of the IPoIB interface to display (e.g., ib0)
Default	N/A	
History	25.02.2002	
Example		

```
admin@nvos:~$ nv show interface ib0
                        operational                                                    applied
----------------------  -------------------------------------------------------------  --------
ip
  arp-timeout           1800                                                           1800
  autoconf              disabled                                                       disabled
  dhcp-client
    state               disabled                                                       disabled
    set-hostname        enabled                                                        enabled
    is-running          no
    has-lease           no
  dhcp-client6
    state               disabled                                                       disabled
    set-hostname        enabled                                                        enabled
    is-running          no
    has-lease           no
  [gateway]
link
  auto-negotiate                                                                     on
  duplex                                                                             full
  speed                                                                              auto
  mac                   80:00:00:02:fe:80:00:00:00:00:00:00:90:0a:84:03:00:76:a8:00
  counters
    in-bytes            72 Bytes
    in-pkts             1
    in-drops            0
    in-errors           0
    out-bytes           480 Bytes
    out-pkts            4
    out-drops           0
    out-errors          0
    carrier-transitions 2
  mtu                   2044                                                           2044
  state                 up                                                             up
ifindex                 8
type                    ipoib                                                          ipoib
```

REST API	GET https://<id>/nvue_v1/interface/{interface-id}
Related commands	nv show interface nv set interface nv unset interface
Notes	

10.2.2.2 nv show interface link

	nv show interface {interface-id} link {state | counters} Displays link information of an IPoIB interface.

Syntax description	interface-id	Name of the interface to display (e.g., ib0).
	state	Show only the data relating to state.
	stats	Show only the data relating to statistics.
Default	N/A	
History	25.02.2002	

Example	<pre>admin@nvos:~$ nv show interface ib0 link state
	operational applied pending
	----------- ------- -------
	down up up</pre>
	<pre>admin@nvos:~$ nv show interface ib0 link counters
	                    operational
	------------------ -----------
	in-bytes 0 Bytes
	in-pkts 0
	in-drops 0
	in-errors 0
	out-bytes 0 Bytes
	out-pkts 0
	out-drops 0
	out-errors 0
	carrier-transitions 1</pre>	

REST API	GET https://<id>/nvue_v1/interface/{interface-id}/link GET https://<id>/nvue_v1/interface/{interface-id}/link/state GET https://<id>/nvue_v1/interface/{interface-id}/link/counters	
Related commands	nv show interface nv set interface link nv unset interface link	
Notes		

10.2.2.3 nv show interface ip

	nv show interface {interface-id} ip {address} Displays IP address of an IPoIB interface.	
Syntax description	interface-id	Name of the interface to display (e.g., ib0)
Default	N/A	
History	25.02.2002	

Example	admin@nvos:~$ nv show interface ib0 ip operational applied pending arp-timeout 1800 1800 1800 autoconf disabled disabled disabled dhcp-client state disabled disabled disabled set-hostname enabled enabled enabled is-running no has-lease no dhcp-client6 state disabled disabled disabled set-hostname enabled enabled enabled is-running no has-lease no [address] 2.2.2.11/24 2.2.2.11/24 2.2.2.11/24 [gateway] admin@nvos:~$ nv show interface ib0 ip address 2.2.2.11/24
REST API	GET https://<id>/nvue_v1/interface/{interface-id}/ip GET https://<id>/nvue_v1/interface/{interface-id}/ip/address
Related commands	nv show interface nv set interface ip nv unset interface ip
Notes	

10.2.2.4 nv set interface link state

	nv set interface {interface-id} link state {value} Sets the administrative link state of a given IPoIB interface.	
Syntax description	interface-id	Name of the interface whose link state to set (e.g., ib0)
	value	New value for the link state: {up, down}
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface ib0 link state up	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/link/state	
Related commands	nv show interface nv unset interface {interface-id} link state	
Notes		

10.2.2.5 nv set/unset interface description

	nv set interface {interface-id} description nv unset interface {interface-id} description Sets the description of a given IPoIB interface. The unset formof the command sets the description of a given IPoIB interface to empty.	
Syntax description	interface-id	Name of the interface whose link state to set (e.g., ib0)

	value	New value for the description.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface ib0 description "IPoIB interface" admin@nvos:~$ nv unset interface ib0 description	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}	
Related commands	nv show interface nvunset interface description	
Notes		

10.2.2.6 nv set/unset interface ip

	nv set interface {interface-id} ip address nv unset interface {interface-id} ip {address {}} Sets the IP address of a given IPoIB interface. The unset form of the command deletes one or more IP addresses assigned to a given IPoIB interface.	
Syntax description	interface-id	Name of the interface whose IP address to set (e.g., ib0)
	ip-prefix-id	IP address and netmask to assign to the interface.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv set interface ib0 ip address 10.10.1.1/8 admin@nvos:~$ nv unset interface ib0 ip address 10.10.1.1/8 admin@nvos:~$ nv unset interface ib0 ip address admin@nvos:~$ nv unset interface ib0 ip	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/ip/address	
Related commands	nv show interface	
Notes		

10.2.2.7 nv unset interface

	nv unset interface {interface-id} Sets all attributes of an IPoIB interface to default values.	
Syntax description	interface-id	Name of the interface to set to default values (e.g., ib0)

Default	N/A
History	25.02.2002
Example	admin@nvos:~$ nv unset interface ib0
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}
Related commands	nv show interface nv set interface
Notes	

10.2.2.8 nv unset interface link

	nv unset interface {interface-id} link {state} Sets the administrative link state of a given IPoIB interface to the default value of "up".	
Syntax description	interface-id	Name of the interface whose link state to set to default (e.g., ib0)
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv unset interface ib0 link admin@nvos:~$ nv unset interface ib0 link state	
REST API	PATCH https://<id>/nvue_v1/interface/{interface-id}/link	
Related commands	nv show interface nv set interface link	
Notes	Since state is the only settable attribute of link, the two forms of the command are equivalent.	

10.3 ibdiagnet

ibdiagnet is an InfiniBand diagnostic net tool that scans fabric by using route packets and then extracts from them all the available information about its connectivity and devices. As a result of running this tool, the log and dump files are produced and saved to the archive file 'ibdiagnet2_output.tgz'.

10.3.1 ibdiagnet Commands

- ibdiagnet Commands

10.3.2 ibdiagnet Commands

- 10.3.2.1 nv show ib ibdiagnet

10.3.2.1 nv show ib ibdiagnet

	nv show ib ibdiagnet Display content of ibdiagnet log file
Syntax Description	N/A
Default	N/A
History	25.02.2002
Example	`admin@nvos:~$ nv show ib ibdiagnet`
REST API	GET https://<ip>/nvue_v1/ib/ibdiagnet
Related Commands	nv show ib ibdiagnet files nv action upload ib ibdiagnet files {archive-name} {remote-url} nv action delete ib ibdiagnet files {archive-name} nv action run ib cmd <ib-cmd-str>
Notes	Log file is generated each time when 'nv action run ib cmd "ibdiagnet"' CLI is invoked.

10.3.2.2 nv show ib ibdiagnet files

	nv show ib ibdiagnet files Display available ibdiagnet archive files in the system
Syntax Description	N/A
Default	N/A
History	25.02.2002

Example	`admin@nvos:~$ nv show ib ibdiagnet files`
	`File name  Ibdiagnet archive file path`
	`---  ---`
	`ibdiagnet2_output.tgz  /host/ibdiagnet/ibdiagnet2_output.tgz`

REST API	GET https://<ip>/nvue_v1/ib/ibdiagnet/files
Related Commands	nv show ib ibdiagnet nv action upload ib ibdiagnet files {archive-name} {remote-url} nv action delete ib ibdiagnet files {archive-name} nv action run ib cmd <ib-cmd-str>
Notes	Archive file is generated each time when 'nv action run ib cmd "ibdiagnet"' CLI is invoked.

10.3.2.3 nv action delete ib ibdiagnet files

	nv action delete ib ibdiagnet files <archive-name> Remove ibdiagnet archive file from file system.	
Syntax Description	archive-name	Name of the ibdiagnet archive file.
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action delete ib ibdiagnet files ibdiagnet2_output.tgz File delete successfully Action succeeded	
REST API	POST https://<ip>/nvue_v1/ib/ibdiagnet/files/{archive-name}	
Related Commands	nv show ib ibdiagnet nv show ib ibdiagnet files nv action upload ib ibdiagnet files {archive-name} {remote-url} nv action run ib cmd <ib-cmd-str>	
Notes	Available ibdiagnet archive files can be viewed via 'nv show ib ibdiagnet files' command.	

10.3.2.4 nv action upload ib ibdiagnet files

	nv action upload ib ibdiagnet files {archive-name} {remote-url} Upload ibdiagnet archive file to remote host.	
Syntax Description	archive-name	Name of the ibdiagnet archive file.
	remote-url	Destination image file name Remote url path to upload a file to. Format: [protocol]://username[:password]@hostname/path/filename Supported protocols: SCP, FTP, SFTP, and HTTPS
Default	N/A	
History	25.02.2002 25.02.4002 Added HTTPS support in remote-url	
Example	admin@nvos:~$ nv action upload ib ibdiagnet files ibdiagnet2_output.tgz scp://username:password@host-name/home/logs/ File upload successfully Action succeeded	
REST API	POST https://<ip>/nvue_v1/ib/ibdiagnet/files/{archive-name}	
Related Commands	nv show ib ibdiagnet nv show ib ibdiagnet files nv action delete ib ibdiagnet files {archive-name} nv action run ib cmd <ib-cmd-str>	
Notes	Available ibdiagnet archive files can be viewed via "nv show ib ibdiagnet files" command.	

10.3.2.5 nv action run ib cmd

	nv action run ib cmd <ib-cmd-str> Run InfiniBand command (e.g., ibdiagnet).	
Syntax Description	ib-cmd-str	String with InfiniBand shell command to run
Default	N/A	
History	25.02.2002	
Example	admin@nvos:~$ nv action run ib cmd "ibdiagnet --get_phy_info" Running ib command: ibdiagnet ibdiagnet output files were archived into ibdiagnet2_output.tgz Action succeeded	
REST API	POST https://<ip>/nvue_v1/ib/cmd -d '{ "@run" : {"state": "start", "parameters": {"cmd" : "ibdiagnet" }} }'	
Related Commands	nv show ib ibdiagnet nv show ib ibdiagnet files nv action upload ib ibdiagnet files {archive-name} {remote-url} nv action delete ib ibdiagnet files {archive-name}	
Notes	Only ibdiagnet command is currently supported in command string.	

10.4 Fabric

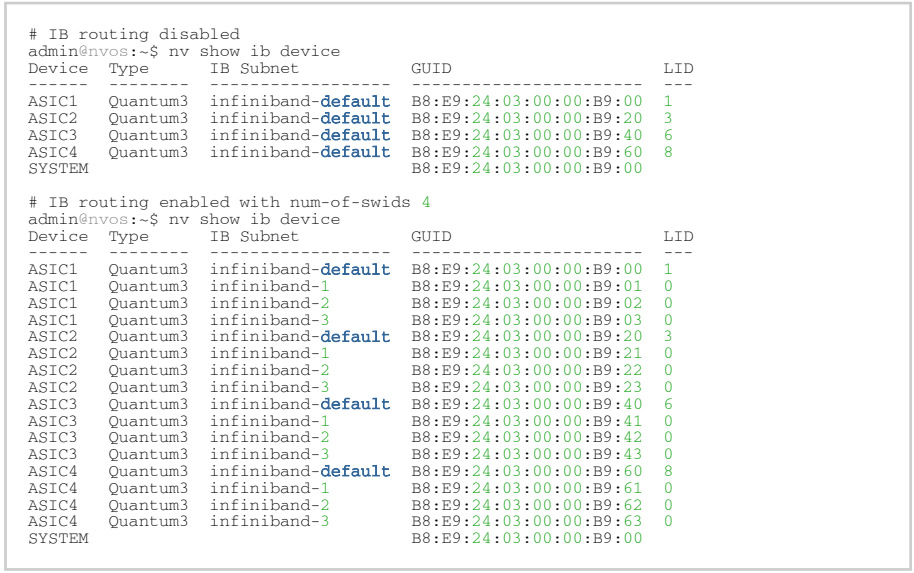
10.4.1 Fabric Commands

- Fabric Commands

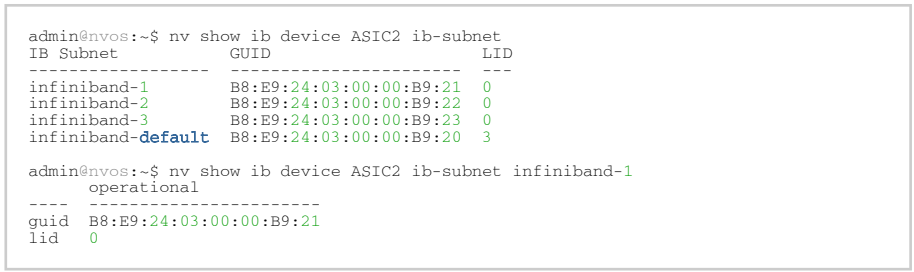
10.4.2 Fabric Commands

10.4.2.1 nv show ib device

	nv show ib device [<device-id>] Shows InfiniBand devices' information.	
Syntax Description	device-id	Optional: show information only for the specified IB device
Default	N/A	
History	25.02.2002 25.02.8008 Updated output and added note	

Example	
REST API	GET https://<id>/nvue_v1/ib/device GET https://<id>/nvue_v1/ib/device/{device-id}
Related Commands	nv action change system profile ib-routing num-of-swids
Notes	By default (IB routing disabled), only the infiniband-default subnet is listed. With IB routing enabled, each ASIC displays a separate GUID and LID entry per subnet.

10.4.2.2 nv show ib device ib-subnet

	nv show ib device <device-id> ib-subnet [<subnet-name>] Shows InfiniBand devices' subnet information.	
Syntax Description	device-id	Specified IB device (e.g., ASIC1, SYSTEM)
	ib-subnet	IB subnet name (e.g., infiniband-default, infiniband-1)
Default	N/A	
History	25.02.8008	
Example		
REST API	GET https://<id>/nvue_v1/ib/device/{device-id}/ib-subnet GET https://<id>/nvue_v1/ib/device/{device-id}/ib-subnet/{ib-subnet-id}	
Related Commands	nv show ib device nv action change system profile ib-routing num-of-swids	
Notes		

10.5 IB Router

InfiniBand (IB) Router provides the ability to send traffic between two or more IB subnets thereby potentially expanding the size of the network to over 40k end-ports, enabling separation, fault resilience and connection to different topologies used by different subnets.

> For more information on IB router architecture and functionality, please refer to the community post IB Router Architecture and Functionality.

10.5.1 Configuring IB Router

10.5.1.1 Prerequisites

1. Platform support: IB Router functionality is supported only on Q3400-RA and Q3401-RD systems.
2. Subnet Manager (SM): Ensure Subnet Managers are not running before configuring the switch as an IB router and assigning interfaces to the IB subnets (SWIDs). Complete all IB router configuration and SWID assignments before starting SMs.
3. Limitations:

> IPoIB: When IB router profile is configured, IPoIB interfaces are not supported and will remain down.

> FNM: External FNM ports are restricted to the default SWID and cannot be assigned to other subnets. Servers should not be connected to the external FNM ports on IB router switches.

10.5.1.2 NVOS Switch Configuration

1. Configure the system profile to enable IB routing with multiple SWIDs:

```
admin@nvos:~$ nv action change system profile ib-routing enabled num-of-swids 4
The operation will reset the system configuration and initiate a reboot.
The resulting system profile values will be:
ib-routing: enabled
num-of-swids: 4
adaptive-routing-groups: 1792
Type [y] to confirm the system profile change and reboot.
Type [N] to abort.
Do you want to continue? [y/N] y
Action executing ...
System will be rebooted, configurations and system files might be deleted
Action executing ...
```

> Profile change triggers a factory reset. All existing configurations will be reset to defaults and the system will reboot.

2. After the reboot, verify the system profile configuration:

```
admin@nvos:~$ nv show system profile
```

```
                        operational
----------------------  -----------
adaptive-routing        enabled
adaptive-routing-groups 1792
breakout-mode           enabled
ib-routing              enabled
num-of-swids            4
```

- Ensure that **ib-routing** is set to **enabled** and that **num-of-swids** matches your configured value.
- After the profile change, all IB interfaces are assigned to the default SWID (**infiniband-default**). To confirm the IB interface assignment to the **infiniband-default** SWID, use the **nv show interface** command.

3. Assign each IB interface to the appropriate IB subnet (SWID) according to the topology port mappings.
   Use the following command format:

```
admin@nvos:~$ nv set interface <interface-id> link ib-subnet <swid-name>
```

   where **<interface-id>** is an interface name or range, and **<swid-name>** is the target SWID.

   Example:

```
admin@nvos:~$ nv set interface sw3-4p1-2 link ib-subnet infiniband-1
admin@nvos:~$ nv config apply -y
Warning: IB interface configuration is changed. In order to apply configuration the port will be restarted.
```

   Ensure Subnet Managers are not running before assigning interfaces to SWIDs. SWID assignments must be completed prior to SM configuration. Changing SWID assignments while Subnet Managers are running may cause inconsistent SM state requiring a reboot of the router.

4. Verify the SWID (IB subnet) assignment for IB interfaces:

```
admin@nvos:~$ nv show interface
Interface  State  Speed  MTU   Type  IB Speed  IB Subnet           Logical State  Physical State
---------  -----  -----  ----  ----  --------  ------------------  -------------  --------------
eth0       up     1G     1500  eth
eth1       up     1G     1500  eth
fnm1       up     800G   4096  fnm   xdr       infiniband-default  Active         LinkUp
sw1p1      up     800G   4096  ib    xdr       infiniband-default  Active         LinkUp
sw1p2      up     800G   4096  ib    xdr       infiniband-default  Active         LinkUp
sw2p1      up     800G   4096  ib    xdr       infiniband-default  Active         LinkUp
sw2p2      up     800G   4096  ib    xdr       infiniband-default  Active         LinkUp
sw3p1      up     800G   4096  ib    xdr       infiniband-1        Active         LinkUp
sw3p2      up     800G   4096  ib    xdr       infiniband-1        Active         LinkUp
sw4p1      up     800G   4096  ib    xdr       infiniband-1        Active         LinkUp
sw4p2      up     800G   4096  ib    xdr       infiniband-1        Active         LinkUp
...
```

10.5.1.3 Verify IB router

After configuring both the IB router and starting Subnet Managers, verify the routing table:

```
admin@nvos:~$ nv show ib router routing-table
IB Subnet           Subnet Prefix             Valid
------------------  ------------------------  -------
infiniband-default  fe:c0:00:00:00:00:00:01   true
infiniband-1        fe:c0:00:00:00:00:00:02   true
infiniband-2        00:00:00:00:00:00:00:00   false
```

```
infiniband-3        00:00:00:00:00:00:00:00  false
```

dd

```
admin@nvos:~$ nv show ib router routing-table
IB Subnet           Subnet Prefix            Valid
------------------  -----------------------  -------
infiniband-default  fe:c0:00:00:00:00:00:01  true
infiniband-1        fe:c0:00:00:00:00:00:02  true
infiniband-2        00:00:00:00:00:00:00:00  false
infiniband-3        00:00:00:00:00:00:00:00  false
```

```
admin@nvos:~$ nv show ib router counters
Counter                      infiniband-default  infiniband-1  infiniband-2  infiniband-3
---------------------------  ------------------  ------------  ------------  ------------
in-bytes                     0                   0             0             0
in-pkts                      0                   0             0             0
in-drops                     0                   0             0             0
out-bytes                    0                   0             0             0
out-pkts                     0                   0             0             0
out-drops                    0                   0             0             0
out-errors                   0                   0             0             0
out-unicast-pkts             0                   0             0             0
out-multicast-pkts           0                   0             0             0
in-unicast-pkts              0                   0             0             0
in-multicast-pkts            0                   0             0             0
buffer-overrun-errors        0                   0             0             0
out-wait                     0                   0             0             0
carrier-down-count           0                   0             0             0
error-recovery               0                   0             0             0
port-rcv-switch-relay-errors 0                   0             0             0
port-rcv-constraint-errors   0                   0             0             0
local-integrity-errors       0                   0             0             0
```

10.5.2 IB Router Commands

- IB Router Commands

10.5.3 IB Router Commands

10.5.3.1 nv action change system profile

	nv action change system profile ib-routing {enabled|disabled} num-of-swids <1-4> [force] Sets the profile of the system.	
Syntax Description	ib-routing	Enables/disables IB routing.
	num-of-swids	Sets the number of SWIDs. Range: 1-4. Valid values are 2-4 when IB routing is enabled, or 1 when IB routing is disabled.
	force	Force the action without asking for user confirmation.
Default	ib-routing: disabled num-of-swids: 1	
History	25.02.7002 25.02.8008 Updated output of the command	
Example	admin@nvos:~$ nv action change system profile ib-routing enabled num-of-swids 2 The operation will reset the system configuration and initiate a reboot. The resulting system profile values will be: ib-routing: enabled num-of-swids: 2 adaptive-routing-groups: 1792 Type [y] to confirm the system profile change and reboot. Type [N] to abort. Do you want to continue? [y/N] y Action executing ... System will be rebooted, configurations and system files might be deleted Action executing ... admin@nvos:~$ nv action change system profile ib-routing disabled num-of-swids 1 The operation will reset the system configuration and initiate a reboot. The resulting system profile values will be: ib-routing: disabled num-of-swids: 1 adaptive-routing-groups: 1792 Type [y] to confirm the system profile change and reboot. Type [N] to abort. Do you want to continue? [y/N] y Action executing ... System will be rebooted, configurations and system files might be deleted Action executing ...	
REST API	POST https://<id>/nvue_v1/system/profile	
Related Commands	nv show system profile	
Notes		

10.5.3.2 nv set/unset interface link ib-subnet

	nv set interface <interface-id> link ib-subnet <subnet-name> nv unset interface <interface-id> link ib-subnet Assigns an IB interface to the specified subnet (SWID). The unset for of the command resets the interface to the default subnet (infiniband-default).	
Syntax Description	interface-id	The IB interface name or range of IB interfaces
	subnet-name	The target subnet name: infiniband-default, infiniband-1... infiniband-3
Default	infiniband-default	

History	25.02.7002 25.02.8008 Updated output of the command
Example	admin@nvos:~$ nv set interface sw3-4p1-2 link ib-subnet infiniband-1 admin@nvos:~$ nv config apply -y Warning: IB interface configuration is changed. In order to apply configuration the port will be restarted.
REST API	PATCH https://<ip>/nvue_v1/interface/<interface-id>/link
Related Commands	nv show interface nv show interface link
Notes	• Supported for IB ports only. FNM ports do not support ib-subnet configuration. • When changing the subnet of an interface, the other port on the same transceiver will be temporarily brought down, even if its subnet is not changing. For example, changing the subnet of `sw1p1` will cause `sw1p2` to be temporarily restarted.

10.5.3.3 nv show ib router routing-table

	nv show ib router routing-table Displays the IB routing table with one entry per subnet, showing the subnet name, subnet prefix, and validity status.
Syntax Description	N/A
Default	N/A
History	25.02.7002
Example	admin@nvos:~$ nv show ib router routing-table IB Subnet Subnet Prefix Valid ---------- ---------- ------- infiniband-1 fec0:0000:0000:0002 true infiniband-2 0000:0000:0000:0000 false infiniband-3 0000:0000:0000:0000 false infiniband-default fec0:0000:0000:0001 true
REST API	GET https://<id>/nvue_v1/ib/router/routing-table
Related Commands	nv change system profile ib-routing num-of-swids nv set interface link ib-subnet nv show ib router ib-subnet
Notes	• For an entry to be valid, at least one port must be assigned to the subnet and in UP state, and a correctly configured SM must be running on that subnet. • Valid `true` indicates that at least one port is assigned to the subnet and in UP state, and the entry was successfully configured by a Subnet Manager.

10.5.3.4 nv show ib router ib-subnet

	nv show ib router ib-subnet Displays detailed information for all subnets
Syntax Description	N/A
Default	N/A

History	25.02.8008
Example	admin@nvos:~$ nv show ib router ib-subnet IB Subnet Subnet Prefix Logical State Physical State GIDs --- infiniband-1 fec0:0000:0000:0002 Active Up fec0:0000:0000:0002:b8e9:2403:0000:b909 fec0:0000:0000:0002:b8e9:2403:0000:b929 fec0:0000:0000:0002:b8e9:2403:0000:b949 fec0:0000:0000:0002:b8e9:2403:0000:b969 infiniband-2 0000:0000:0000:0000 Initialize Up 0000:0000:0000:0000:b8e9:2403:0000:b90a 0000:0000:0000:0000:b8e9:2403:0000:b92a 0000:0000:0000:0000:b8e9:2403:0000:b94a 0000:0000:0000:0000:b8e9:2403:0000:b96a infiniband-3 0000:0000:0000:0000 Initialize Up 0000:0000:0000:0000:b8e9:2403:0000:b90b 0000:0000:0000:0000:b8e9:2403:0000:b92b 0000:0000:0000:0000:b8e9:2403:0000:b94b 0000:0000:0000:0000:b8e9:2403:0000:b96b infiniband-default fec0:0000:0000:0001 Active Up fe80:0000:0000:0000:b8e9:2403:0000:b908 fe80:0000:0000:0000:b8e9:2403:0000:b928 fe80:0000:0000:0000:b8e9:2403:0000:b948 fe80:0000:0000:0000:b8e9:2403:0000:b968
REST API	GET https://<id>/nvue_v1/ib/router/ib-subnet
Related Commands	nv change system profile ib-routing num-of-swids nv set interface link ib-subnet nv show ib device nv show ib device ib-subnet
Notes	

10.5.3.5 nv show ib router ib-subnet subnet-name

	nv show ib router ib-subnet <subnet-name> Displays detailed information for a single IB subnet.	
Syntax Description	subnet-name	The target subnet name: infiniband-default, infiniband-1... infiniband-3
Default	N/A	
History	25.02.8008	
Example	admin@nvos:~$ nv show ib router ib-subnet infiniband-1 operational --- subnet-prefix fec0:0000:0000:0002 logical-state Active physical-state Up [gid] fec0:0000:0000:0002:b8e9:2403:0000:b909 [gid] fec0:0000:0000:0002:b8e9:2403:0000:b929 [gid] fec0:0000:0000:0002:b8e9:2403:0000:b949 [gid] fec0:0000:0000:0002:b8e9:2403:0000:b969	
REST API	GET https://<id>/nvue_v1/ib/router/ib-subnet/{ib-subnet-id}	

Related Commands	nv change system profile ib-routing num-of-swids nv set interface link ib-subnet nv show ib router ib-subnet nv show ib device nv show ib device ib-subnet
Notes	

10.5.3.6 nv show ib router ib-subnet subnet-name counters

	nv show ib router ib-subnet <subnet-name> counters Displays counters for a single IB subnet.	
Syntax Description	subnet-name	The target subnet name: infiniband-default, infiniband-1... infiniband-3
Default	N/A	
History	25.02.8008	
Example	admin@nvos:~$ nv show ib router ib-subnet infiniband-1 counters operational --- in-bytes 0 Bytes in-pkts 0 in-drops 0 out-bytes 0 Bytes out-pkts 0 out-drops 0 out-errors 0 out-unicast-pkts 0 out-multicast-pkts 0 in-unicast-pkts 0 in-multicast-pkts 0 buffer-overrun-errors 0 out-wait 0 link carrier-down-count 0 error-recovery 0 port-rcv-switch-relay-errors 0 port-rcv-constraint-errors 0 local-integrity-errors 0	
REST API	GET https://<id>/nvue_v1/ib/router/ib-subnet/{ib-subnet-id}/counters	
Related Commands	nv change system profile ib-routing num-of-swids nv set interface link ib-subnet	
Notes	`in-errors` are not supported (compared to IB port counters).	

10.5.3.7 nv show ib router ib-subnet subnet-name counters link

	nv show ib router ib-subnet <subnet-name> counters link Displays link counters for a single IB subnet.	
Syntax Description	subnet-name	The target subnet name: infiniband-default, infiniband-1... infiniband-3
Default	N/A	
History	25.02.8008	

Example	admin@nvos:~$ nv show ib router ib-subnet infiniband-1 counters link operational --- carrier-down-count 0 error-recovery 0 port-rcv-switch-relay-errors 0 port-rcv-constraint-errors 0 local-integrity-errors 0
REST API	GET https://<id>/nvue_v1/ib/router/ib-subnet/{ib-subnet-id}/counters/link
Related Commands	nv change system profile ib-routing num-of-swids nv set interface link ib-subnet nv show ib router ib-subnet counters
Notes	port-rcv-remote-physical-errors are not supported (compared to IB port counters).

10.5.3.8 nv show ib router ib-subnet subnet-name gid

	nv show ib router ib-subnet <subnet-name> gid Displays GIDs of each ASIC for a single IB subnet.	
Syntax Description	subnet-name	The target subnet name: infiniband-default, infiniband-1... infiniband-3
Default	N/A	
History	25.02.8008	
Example	admin@nvos:~$ nv show ib router ib-subnet infiniband-1 gid GID ASIC --- fec0:0000:0000:0002:b8e9:2403:0000:b909 ASIC1 fec0:0000:0000:0002:b8e9:2403:0000:b929 ASIC2 fec0:0000:0000:0002:b8e9:2403:0000:b949 ASIC3 fec0:0000:0000:0002:b8e9:2403:0000:b969 ASIC4	
REST API	GET https://<id>/nvue_v1/ib/router/ib-subnet/{ib-subnet-id}/gid	
Related Commands	nv change system profile ib-routing num-of-swids nv set interface link ib-subnet nv show ib router ib-subnet	
Notes		

10.5.3.9 nv show ib router counters

	nv show ib router counters Displays counters for all IB subnets.
Syntax Description	N/A
Default	N/A
History	25.02.8008

Example	admin@nvos:~$ nv show ib router --view counters Counter infiniband-default infiniband-1 infiniband-2 infiniband-3 in-bytes 0 0 0 0 in-pkts 0 0 0 0 in-drops 0 0 0 0 out-bytes 0 0 0 0 out-pkts 0 0 0 0 out-drops 0 0 0 0 out-errors 0 0 0 0 out-unicast-pkts 0 0 0 0 out-multicast-pkts 0 0 0 0 in-unicast-pkts 0 0 0 0 in-multicast-pkts 0 0 0 0 buffer-overrun-errors 0 0 0 0 out-wait 0 0 0 0 carrier-down-count 0 0 0 0 error-recovery 0 0 0 0 port-rcv-switch-relay-errors 0 0 0 0 port-rcv-constraint-errors 0 0 0 0 local-integrity-errors 0 0 0 0
REST API	GET https://<id>/nvue_v1/ib/router
Related Commands	nv change system profile ib-routing num-of-swids nv set interface link ib-subnet nv show ib router ib-subnet subnet-name counters nv show ib router ib-subnet subnet-name counters link
Notes	

10.5.3.10 nv action clear ib router counters

	nv action clear ib router counters Clear counters for all IB subnets.
Syntax Description	N/A
Default	N/A
History	25.02.8008
Example	admin@nvos:~$ nv action clear ib router counters Action executing ... Cleared IB router counters successfully Action succeeded
REST API	POST https://<id>/nvue_v1/ib/router
Related Commands	nv change system profile ib-routing num-of-swids nv set interface link ib-subnet nv show ib router ib-subnet subnet-name counters nv show ib router ib-subnet subnet-name counters link nv show ib router counters
Notes	

10.6 HCA Peer Port Telemetry

- 10.6.5 HCA Telemetry Commands

NVOS enables users to monitor telemetry for Host Channel Adapter (HCA) ports that are directly connected to the switch, without requiring access to the HCA itself.

Telemetry is streamed only for HCA ports with an active link. The available metrics align with the counters exposed for Infiniband interfaces and include:

- Traffic statistics
- Link status
- Infiniband data
- PHY diagnostics

10.6.1 How It Works

The HCA Peer Port Telemetry feature state can be set over NVOS using simple NVUE CLI commands:

Show command:

```
nvos@switch:~$ nv show system peer-port
          operational  applied
--------  -----------  -------
state     disabled     disabled
interval  5            5
```

Set command:

```
nvos@switch:~$ nv set system peer-port state <enabled | disabled>
```

Unset command:

```
nvos@switch:~$ nv unset system peer-port state
```

The state is **disabled by default** and the unset command will restore the state to **disabled**, if it is not already.

- By default, data is collected every **5 seconds**.
- Each discovered HCA port with an active link appears as a **peer-port** (hcaA2p1, hcaA2p2, and so on).
- When an HCA link goes down, its peer-port entry is automatically removed from the output.

Telemetry data can be accessed using:

- The **nv show peer-port** CLI command hierarchy
- gNMI

10.6.2 Viewing HCA Peer-Ports

To view all active HCA peer-ports and their topology information, run the following command:

```
admin@nvos:~$ nv show peer-port
Peer Port   Port GUID          Node GUID          HCA Node Description          HCA Port Index  Switch Port
----------  -----------------  -----------------  ----------------------------  --------------  -----------
hcaA2p1     0x90e3170300cb373c  0x90e3170300cb373c  fit-nvos-vrt-mec-161-002 smi2  1               swA2p1
hcaA2p1pl1  0x90e3170300cb373c  0x90e3170300cb373c  fit-nvos-vrt-mec-161-002 smi2  1               swA2p1pl1
```

```
hcaA2p1pl2  0x90e3170300cb373c  0x90e3170300cb373c  fit-nvos-vrt-mec-161-002 smi2  2              swA2p1pl2
hcaA2p1pl3  0x90e3170300cb373c  0x90e3170300cb373c  fit-nvos-vrt-mec-161-002 smi2  3              swA2p1pl3
hcaA2p1pl4  0x90e3170300cb373c  0x90e3170300cb373c  fit-nvos-vrt-mec-161-002 smi2  4              swA2p1pl4
hcaA2p2     0x90e3170300cb3f3c  0x90e3170300cb3f3c  fit-nvos-vrt-mec-161-003 smi0  1              swA2p2
hcaA2p2pl1  0x90e3170300cb3f3c  0x90e3170300cb3f3c  fit-nvos-vrt-mec-161-003 smi0  1              swA2p2pl1
hcaA2p2pl2  0x90e3170300cb3f3c  0x90e3170300cb3f3c  fit-nvos-vrt-mec-161-003 smi0  2              swA2p2pl2
hcaA2p2pl3  0x90e3170300cb3f3c  0x90e3170300cb3f3c  fit-nvos-vrt-mec-161-003 smi0  3              swA2p2pl3
hcaA2p2pl4  0x90e3170300cb3f3c  0x90e3170300cb3f3c  fit-nvos-vrt-mec-161-003 smi0  4              swA2p2pl4
```

10.6.3 Viewing a Specific HCA Port

To inspect a specific HCA port, run the following command:

```
admin@nvos:~$ nv show peer-port hcaA2p1
```

10.6.4 Viewing Counters and Link Health

To view counters, run the following command:

```
admin@nvos:~$ nv show peer-port hcaA2p1 counters
```

To view link information, run the following command:

```
admin@nvos:~$ nv show peer-port hcaA2p1 link
```

See the HCA Peer Port Telemetry Commands section below for full details and output examples.

⚠ Limitations

- When an HCA link transitions up or down, it may take up to 1 minute for the change to appear in `nv show peer-port`.
- PHY lane statistics are reported in `nv show peer-port <peer-port-id> link phy health`.

10.6.5 HCA Telemetry Commands

10.6.6 HCA Peer Port Telemetry Commands

10.6.6.1 nv show system peer-port

	nv show system peer-port Display the settings of HCA peer-port configuration.	
Syntax Description	N/A	
Default	N/A	
History	25.03.10xx	
Example	admin@nvos:~$ nv show system peer-port operational applied -------- ----------- -------- state disabled disabled interval 5 5	
REST API	GET https://<ip>/nvue_v1/system/peer-port	
Related Commands	nv set system peer-port state {enabled|disabled}	
Notes		

10.6.6.2 nv set/unset system peer-port state

	nv set system peer-port state {enabled | disabled} nv unset system peer-port state Enable/disable HCA peer-port functionality on a switch. The unset form of the command sets peer-port functionality to default.	
Syntax Description	enabled	Enable HCA peer-port functionality
	disabled	Disable HCA peer-port functionality
Default	Disabled	
History	25.03.10xx	
Example	admin@nvos:~$ nv set system peer-port state enabled admin@nvos:~$ nv config apply	
REST API	PATCH https://<ip>/nvue_v1/system/peer-port	
Related Commands	nv show system peer-port	
Notes		

10.6.6.3 nv set/unset system peer-port interval

	nv set system peer-port interval {integer:1 - 3600} nv unset system peer-port interval Set the HCA peer-port telemetry polling interval in seconds	
Syntax Description	interval	The HCA peer-port telemetry polling interval in seconds

Default	5	
History	25.03.10xx	
Example	admin@nvos:~$ nv set system peer-port interval 10 admin@nvos:~$ nv config apply	
REST API	PATCH https://<ip>/nvue_v1/system/peer-port	
Related Commands	nv show system peer-port	
Notes		

10.6.6.4 nv show peer-port

	nv show peer-port Display information about all active peer ports in the system.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	
Example	admin@nvos:~$ nv show peer-port Peer Port Switch Port Port GUID Node GUID HCA Node Description HCA Port Index ---------- ---------- ---------- ---------- ---------- hca50p1 1 0xc470bd0300a99d88 sw50p1 0xc470bd0300a99d88 fit-nos-vrt-40-045 HCA-1 hca50p1pl1 1 0xc470bd0300a99d88 sw50p1pl1 0xc470bd0300a99d88 fit-nos-vrt-40-045 HCA-1 hca50p1pl2 2 0xc470bd0300a99d88 sw50p1pl2 0xc470bd0300a99d88 fit-nos-vrt-40-045 HCA-1 hca50p1pl3 3 0xc470bd0300a99d88 sw50p1pl3 0xc470bd0300a99d88 fit-nos-vrt-40-045 HCA-1 hca50p1pl4 4 0xc470bd0300a99d88 sw50p1pl4 0xc470bd0300a99d88 fit-nos-vrt-40-045 HCA-1	
REST API	GET https://<ip>/nvue_v1/peer-port	
Related Commands	nv show peer-port id nv show peer-port id counters nv show peer-port id link	
Notes		

10.6.6.5 nv show peer-port id

	nv show peer-port <peer-port-id> Display information about a single peer port in the system.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	

Example	admin@nvos:~$ nv show peer-port hca50p1 operational --- node-guid 0xc470bd0300a99d88 port-guid 0xc470bd0300a99d88 associated-switch-tray-number 0 associated-switch-asic-in-tray 1 associated-switch-port sw50p1 chassis-sn MT2443X00270 peer-port-name hca50p1 host-name fit-nos-vrt-40-045 hca-node-description fit-nos-vrt-40-045 HCA-1 hca-port-index 1 hca-ib-port 1 counters in-bytes 446.74 MB in-pkts 1626587 in-drops 0 in-errors 0 out-bytes 446.73 MB out-pkts 1626498 out-drops 1 out-errors 0 out-unicast-pkts 1626498 out-multicast-pkts 0 in-unicast-pkts 1626486 in-multicast-pkts 101 buffer-overrun-errors 0 out-wait 0 link carrier-down-count 510 error-recovery 0 port-rcv-remote-physical-errors 0 port-rcv-switch-relay-errors 0 port-rcv-constraint-errors 0 local-integrity-errors 0 ib [errors] icrc-errors [errors] symbol-errors [errors] tx-parity-errors [drops] qp1-drops
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}
Related Commands	nv show peer-port nv show peer-port id counters nv show peer-port id link
Notes	

10.6.6.6 nv show peer-port id counters

	nv show peer-port <peer-port-id> counters Display counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	

Example	admin@nvos:~$ nv show peer-port hca50p1 counters operational in-bytes 468630004 Bytes in-pkts 1627229 in-drops 0 in-errors 0 out-bytes 468616320 Bytes out-pkts 1627140 out-drops 1 out-errors 0 out-unicast-pkts 1627140 out-multicast-pkts 0 in-unicast-pkts 1627128 in-multicast-pkts 101 buffer-overrun-errors 0 out-wait 0 link carrier-down-count 510 error-recovery 0 port-rcv-remote-physical-errors 0 port-rcv-switch-relay-errors 0 port-rcv-constraint-errors 0 local-integrity-errors 0 Error Counters Counter Receive Transmit icrc-errors 0 n/a symbol-errors 0 n/a tx-parity-errors 0 0 Drop Counters Counter Receive Transmit qp1-drops 0 0 Fast Recovery Counters Trigger Consecutive Normal Windows Warning Events Error Events
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/counters
Related Commands	nv show peer-port nv show peer-port id nv show peer-port id link
Notes	

10.6.6.7 nv show peer-port id counters ib

	nv show peer-port <peer-port-id> counters ib Display IB counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	

Example	admin@nvos:~$ nv show peer-port hca50p1 counters ib Drop Counters Counter Receive Transmit qp1-drops 0 0 Error Counters Counter Receive Transmit icrc-errors 0 n/a symbol-errors 0 n/a tx-parity-errors 0 0 Fast Recovery Counters No Data
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/counters/ib
Related Commands	nv show peer-port id counters
Notes	

10.6.6.8 nv show peer-port id counters ib drops

	nv show peer-port <peer-port-id> counters ib drops Display IB drop counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	
Example	admin@nvos:~$ nv show peer-port hca50p1 counters ib drops Counter Receive Transmit qp1-drops 0 0	
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/counters/ib/drops	
Related Commands	nv show peer-port id counters nv show peer-port id counters ib	
Notes		

10.6.6.9 nv show peer-port id counters ib errors

	nv show peer-port <peer-port-id> counters ib errors Display IB errors counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	

Example	admin@nvos:~$ nv show peer-port hca50p1 counters ib errors Counter Receive Transmit ---------- ------- -------- icrc-errors 0 n/a symbol-errors 0 n/a tx-parity-errors 0 0
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/counters/ib/errors
Related Commands	nv show peer-port id counters nv show peer-port id counters ib
Notes	

10.6.6.10 nv show peer-port id counters link

	nv show peer-port <peer-port-id> counters link Display link counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	
Example	admin@nvos:~$ nv show peer-port hca50p1 counters link operational ---------- ---------- carrier-down-count 0 error-recovery 0 port-rcv-remote-physical-errors 0 port-rcv-switch-relay-errors 0 port-rcv-constraint-errors 0 local-integrity-errors 0	
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/counters/link	
Related Commands	nv show peer-port id counters	
Notes		

10.6.6.11 nv show peer-port id link

	nv show peer-port <peer-port-id> link Display link PHY counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	

Example	admin@nvos:~$ nv show peer-port hca50p1 link operational physical-state LinkUp logical-state Active speed 800G width 4X phy health [lane] 0 [lane] 1 [lane] 2 [lane] 3 [lane] 4 [lane] 5 [lane] 6 [lane] 7 histogram [rs-fec-corrected-errors] 0 [rs-fec-corrected-errors] 1 [rs-fec-corrected-errors] 2 [rs-fec-corrected-errors] 3 [rs-fec-corrected-errors] 4 [rs-fec-corrected-errors] 5 [rs-fec-corrected-errors] 6 [rs-fec-corrected-errors] 7 [rs-fec-corrected-errors] 8 [rs-fec-corrected-errors] 9 [rs-fec-corrected-errors] 10 [rs-fec-corrected-errors] 11 [rs-fec-corrected-errors] 12 [rs-fec-corrected-errors] 13 [rs-fec-corrected-errors] 14 [rs-fec-corrected-errors] 15 time-since-last-clear-min 221.756 symbol-errors 0 symbol-ber 1.5e-254 raw-ber 1e-07 phy-received-bits 11309877886954920 effective-errors 0 effective-ber 1.5e-254 detail profile-fec-in-use Int_KP4_FEC phy-manager-state Active_or_Linkup sync-header-error-counter 0 plr-rcv-codes 0 plr-rcv-codes-err 0 plr-rcv-uncorrectable-code 0 plr-xmit-codes 0 plr-xmit-retry-codes 0 plr-xmit-retry-events 0 plr-sync-events 0 plr-codes-loss 0 plr-bw-loss-percent 0.0 rq-general-error 0
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/link
Related Commands	nv show peer-port nv show peer-port id
Notes	

10.6.6.12 nv show peer-port id link phy

	nv show peer-port <peer-port-id> link phy Display PHY counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	

Example	admin@nvos:~$ nv show peer-port hca50p1 link phy operational health [lane] 0 [lane] 1 [lane] 2 [lane] 3 [lane] 4 [lane] 5 [lane] 6 [lane] 7 histogram [rs-fec-corrected-errors] 0 [rs-fec-corrected-errors] 1 [rs-fec-corrected-errors] 2 [rs-fec-corrected-errors] 3 [rs-fec-corrected-errors] 4 [rs-fec-corrected-errors] 5 [rs-fec-corrected-errors] 6 [rs-fec-corrected-errors] 7 [rs-fec-corrected-errors] 8 [rs-fec-corrected-errors] 9 [rs-fec-corrected-errors] 10 [rs-fec-corrected-errors] 11 [rs-fec-corrected-errors] 12 [rs-fec-corrected-errors] 13 [rs-fec-corrected-errors] 14 [rs-fec-corrected-errors] 15 time-since-last-clear-min 222.172 symbol-errors 0 symbol-ber 1.5e-254 raw-ber 1e-07 phy-received-bits 11331129161166360 effective-errors 0 effective-ber 1.5e-254 detail profile-fec-in-use Int_KP4_FEC phy-manager-state Active_or_Linkup sync-header-error-counter 0 plr-rcv-codes 0 plr-rcv-codes-err 0 plr-rcv-uncorrectable-code 0 plr-xmit-codes 0 plr-xmit-retry-codes 0 plr-xmit-retry-events 0 plr-sync-events 0 plr-codes-loss 0 plr-bw-loss-percent 0.0 rq-general-error 0
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/link/phy
Related Commands	nv show peer-port id link
Notes	

10.6.6.13 nv show peer-port id link phy health

	nv show peer-port <peer-port-id> link phy health Display PHY health counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	

Example	admin@nvos:~$ nv show peer-port hca50p1 link phy health operational time-since-last-clear-min 222.588 symbol-errors 0 symbol-ber 1.5e-254 raw-ber 1e-07 phy-received-bits 11352381187907760 effective-errors 0 effective-ber 1.5e-254 Lane Stats Lane raw-ber phy-raw-errors 0 0 0 1 0 0 2 0 0 3 0 0 4 None None 5 None None 6 None None 7 None None
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/link/phy/health
Related Commands	nv show peer-port id link nv show peer-port id link phy
Notes	

10.6.6.14 nv show peer-port id link phy health histogram

	nv show peer-port <peer-port-id> link phy health histogram Display PHY health histogram counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	
Example	admin@nvos:~$ nv show peer-port hca50p1 link phy health histogram rs-fec-corrected-errors Bin count 0 0 1 0 2 0 3 0 4 0 5 0 6 0 7 0 8 0 9 0 10 0 11 0 12 0 13 0 14 0 15 0	
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/link/phy/health/histogram	
Related Commands	nv show peer-port id link nv show peer-port id link phy nv show peer-port id link phy health	
Notes		

10.6.6.15 nv show peer-port id link phy detail

	nv show peer-port <peer-port-id> link phy detail Display PHY detail counters of a peer port.	
Syntax Description	peer-port-id	The name of the peer HCA port (e.g., hcaA2p1, hcaA2p2)
Default	N/A	
History	25.03.10xx	
Example	admin@nvos:~$ nv show peer-port hca50p1 link phy detail operational profile-fec-in-use Int_KP4_FEC phy-manager-state Active_or_Linkup sync-header-error-counter 0 plr-rcv-codes 0 plr-rcv-codes-err 0 plr-rcv-uncorrectable-code 0 plr-xmit-codes 0 plr-xmit-retry-codes 0 plr-xmit-retry-events 0 plr-sync-events 0 plr-codes-loss 0 plr-bw-loss-percent 0.0 rq-general-error 0	
REST API	GET https://<ip>/nvue_v1/peer-port/{peer-port-id}/link/phy/detail	
Related Commands	nv show peer-port id link nv show peer-port id link phy	
Notes		

10.7 Plane-Port Telemetry

NVOS enables users to monitor the telemetry for the individual plane ports that constitute an aggregate port.

10.7.1 How It Works

The InfiniBand plane-port telemetry feature can be enabled or disabled in NVOS using NVUE CLI commands.

Show command:

```
nvos@switch:~$ nv show system plane-port
         operational  applied
--------  -----------  -------
state     disabled     disabled
```

Set command:

```
nvos@switch:~$ nv set system plane-port state <enabled | disabled>
```

Unset command:

```
nvos@switch:~$ nv unset system plane-port state
```

The feature is disabled by default. The unset command restores the state to **disabled** if it is not already disabled.

Plane-port telemetry data can be accessed through:

- The **nv show interface** CLI command hierarchy
- gNMI

10.7.2 Viewing Plane-Ports

When plane-port support is enabled, plane ports appear as read-only entities alongside aggregate ports in both NVUE and gNMI output. The **type** field is set to **ibpp**, indicating an InfiniBand plane port.

For example:

```
admin@nvos:~$ nv show interface
Interface  State  Speed  MTU    Type      Description  IB Speed  IB Subnet           Logical State  Physical State
Summary
---------  -----  -----  -----  --------  -----------  --------  ------------------  -------------
-------------------------  ------------------------------------------------------
sw1p1      down                 ib                                 infiniband-default  Down           Disabled
sw1p1pl1   down                 ibpp                               infiniband-default  Down           Disabled
sw1p1pl2   down                 ibpp                               infiniband-default  Down           Disabled
sw1p1pl3   down                 ibpp                               infiniband-default  Down           Disabled
sw1p1pl4   down                 ibpp                               infiniband-default  Down           Disabled
sw1p2      up     800G   4096   ib                         xdr     infiniband-default  Active         LinkUp
sw1p2pl1   up     200G   4096   ibpp                       xdr     infiniband-default  Active         LinkUp
sw1p2pl2   up     200G   4096   ibpp                       xdr     infiniband-default  Active         LinkUp
sw1p2pl3   up     200G   4096   ibpp                       xdr     infiniband-default  Active         LinkUp
sw1p2pl4   up     200G   4096   ibpp                       xdr     infiniband-default  Active         LinkUp
...
```

All existing **nv show interface** commands also apply to plane ports.

10.7.3 Viewing a Specific Plane-Port

To inspect a specific plane-port, run the following command:

```
admin@nvos:~$ nv show interface sw1p1pl1
```

10.7.4 Viewing Counters and Link Health

To view counters, run the following command:

```
admin@nvos:~$ nv show interface sw1p1pl1 counters
```

To view link information, run the following command:

```
admin@nvos:~$ nv show interface sw1p1pl1 link
```

See the Plane-Port Telemetry Commands section below for full details and output examples.

10.7.5 Plane-Port Telemetry Commands

10.7.6 Plane-Port Telemetry Commands

10.7.6.1 nv show system plane-port

	nv show system plane-port Display the settings of plane-port configuration.
Syntax Description	N/A
Default	N/A
History	25.03.10xx

Example	`nv show system plane-port`
	`operational applied`
	`----- ----------- -------`
	`state disabled disabled`

REST API	GET https://<ip>/nvue_v1/system/plane-port
Related Commands	nv set system plane-port state {enabled|disabled}
Notes	

10.7.6.2 nv set/unset system plane-port state

	nv set system plane-port state {enabled | disabled} nv unset system plane-port state Enable/disable plane-port functionality on a switch. The unset form of the command sets plane-port functionality to default.	
Syntax Description	enabled	Enable plane-port functionality
	disabled	Disable plane-port functionality
Default	Disabled	
History	25.03.10xx	
Example	`admin@nvos:~$ nv set system plane-port state enabled` `admin@nvos:~$ nv config apply`	
REST API	PATCH https://<ip>/nvue_v1/system/plane-port	
Related Commands	nv show system plane-port	
Notes		

11 Appendix—Moving from MLNX-OS to NVOS

To install NVOS, you use ONIE (Open Network Install Environment), an extension to the traditional U-Boot software that allows for automatic discovery of a network installer image. This facilitates the ecosystem model of procuring switches with an operating system choice, such as NVOS. The easiest way to install NVOS with ONIE is with local HTTP discovery:

1. If your host (laptop or server) is IPv6-enabled, make sure it is running a web server. If your host is IPv4-enabled, make sure it is running DHCP in addition to a web server.
2. Download the NVOS installation file to the root directory of the web server. Rename this file `onie-installer`.
3. Connect your host using an Ethernet cable to the management Ethernet port of the switch.
4. Power on the switch. The switch downloads the ONIE image installer and boots. You can watch the installation progress in your terminal. After the installation completes, the NVOS login prompt appears in the terminal window.

> ⚠ These steps describe a flexible unattended installation method; you do not need a console cable. A fresh install with ONIE using a local web server typically completes in less than ten minutes. However, you have more options for installing NVOS with ONIE, such as using a local file, FTP or USB. See Installing a New NVOS Image for more options.

After installing NVOS, you are ready to:

- Log in to NVOS on the switch.
- Configure NVOS. This quick start guide provides instructions on configuring switch ports interface.

12 Appendix—Design Consideration Using Q3200

12.1 Motivation for Using Q3200

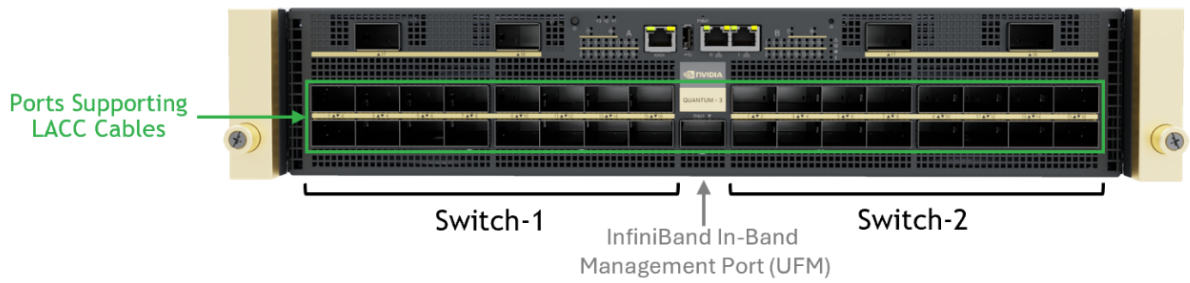
The NVIDIA Quantum-X800 Q3200 is an ideal InfiniBand switching solution for mixed-generation environments, where data-center infrastructures need to integrate modern high-speed networking (Quantum-X800 based) with prior-generation systems.
It incorporates two fully independent switches in a compact 2U air-cooled chassis, each providing 36 ports of 800 Gb/s.

The Q3200 is commonly deployed in storage-rich architectures thanks to its dense port configuration and high aggregate bandwidth. It is particularly effective for environments that require:

- High-speed connectivity for new compute clusters
- Seamless integration with HDR, NDR, NDR200, and XDR400 InfiniBand systems
- Robust networking to large-scale storage infrastructures

Overall, the Q3200 enables flexible, high-throughput fabrics while simplifying mixed-generation deployments.

12.2 Q3200 Hardware Overview



12.2.1 Dual Independent Switch Architecture

- Two independent 36-port switches in a single 2U enclosure
- 18 OSFP cages per switch, each supporting 2x 800G ports
- 36 × 800G ports per switch
- Air-cooled design
- AC power
- Managed switch with NVOS support

12.2.2 Management Interface

- Dedicated InfiniBand in-band management port (FNM port), for both internal switches
- FNM port connectivity to UFM at NDR200 speed

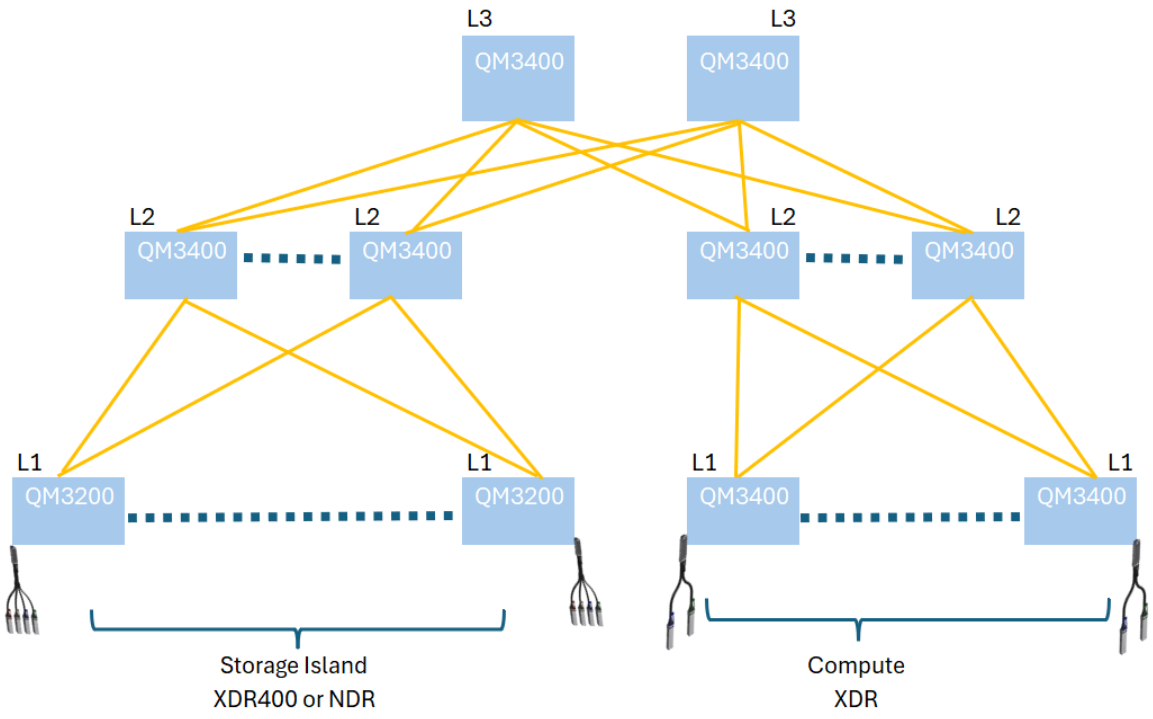
12.3 Switch Radix Capability

Switch Radix Capability		
Speed	**Max Ports per Switch**	**Max Ports per Q3200-RA**
XDR	36	72
XDR400 (2 lanes)	72	144
NDR	36	72
NDR200 (2 lanes)	72	144
HDR	36	72

12.4 Connectivity to Storage

The diagram below illustrates a Q3200 deployed in the leaf layer, connecting upstream to the Q3400 spine and downstream to the compute layer via XDR400 or NDR links. The Q3200 connects to the Q3400 at XDR speed, while connections to the servers operate at XDR400 or NDR speeds. Each 1.6 Tb/s OSFP port on the Q3200 is split into four in cases of XDR400 (or NDR200) links to support server connectivity.

Compute and Storage Fabric with East-West 800Gb/s and 400Gb/s Connectivity



12.5 XDR400 Two-Port Split Option

Q3200 ports can be configured to split into two XDR400 ports. In this mode, each resulting port uses two lanes of 200 Gb/s, providing 2 × 400 Gb/s connectivity from a single physical port.

To configure this split mode, use the following NVOS commands:

```
nv set interface swB2p1-2 link connection-mode ndr
nv set interface swB2p1-2 link breakout 2x-xdr
nv config apply -y
nv config save
```

This split option is useful when connecting Q3200 ports to systems while preserving efficient use of the switch's high-speed OSFP interfaces.

12.6 Port Behavior Limitation

The following limitations apply to ports 17, 18, 35, and 36:

- Split mode is not supported
- Linear Active Copper Cable (LACC) is not supported

As a result, the XDR400 two-port split option described above cannot be applied to these ports.

12.7 Release Notes Reference

For details on known issues, limitations, and recommended workarounds for Q3200 deployments, refer to the NVIDIA NVOS Release Notes available in the Enterprise Support Portal.

13 Appendix—Switch SSD Firmware Update Customer Guide

This document describes a rare issue where the system SSD (NVMe device) becomes unresponsive during normal operation while traffic continues to run. It outlines the symptoms, affected components, and provides a detailed procedure for installing a firmware fix through the regular NVOS update mechanism.

13.1 Issue Description

During normal operation, the system SSD (NVMe device) may become unresponsive even while traffic continues to flow. When this happens, the operating system (NVOS) error logs will resemble the following output:

```
WARNING kernel: [2729682.680668] nvme nvme0: I/O 132 QID 3 timeout, aborting
WARNING kernel: [2729682.682007] nvme nvme0: Abort status: 0x0
WARNING kernel: [2729713.399278] nvme nvme0: I/O 132 QID 3 timeout, reset controller
WARNING kernel: [2729774.836562] nvme nvme0: I/O 21 QID 0 timeout, reset controller
ERR kernel: [2729795.339632] nvme nvme0: Device not ready; aborting reset, CSTS=0x1
WARNING kernel: [2729815.909129] nvme nvme0: Removing after probe failure status: -19
...
EXT4-fs error (device nvme0n1p3): ext4_get_inode_loc:4513: inode #531990: block 2097665: comm healthd: unable to
read itable block
EXT4-fs error (device nvme0n1p3): ext4_get_inode_loc:4513: inode #534476: block 2097820: comm python3: unable to
read itable block
```

13.2 Affected Components

Affected Part Number: VTPM24CEXI080-BM1 10006 (MEM000490)

Affected Firmware: CE00A450, CE00A400

Solution: Issue is fixed in the new SSD firmware version CE00A474

13.3 SSD Firmware Update Methods

NVOS supports two options for updating SSD firmware:

1. Automatic installation of the SSD firmware as part of the NVOS upgrade—available from version 25.02.80xx onward.
2. Manual installation using the NVUE CLI commands — available from version 25.02.80xx onward.

13.4 Auto SSD Firmware Installation

In NVOS systems, **SSD (NVMe) firmware is automatically installed during the standard NVOS software upgrade process**. No special user action is required.
The updated SSD firmware becomes active **after the system completes its routine power cycles/cold reboot**, which occur naturally as part of the NVOS upgrade flow.

13.4.1 Verification

After the NVOS update and power cycle/cold reboot of the system, do the following:

1. Log in to the system.
2. Run the following command to check the SSD firmware version:

```
admin@nvos:~$ nv show platform firmware SSD
```

Expected output:

```
                  operational
part-number       Virtium VTPM24CEXI080-BM110006
actual-firmware   CE00A474
fw-source         N/A
```

- **actual-firmware** should show **CE00A474** (indicating the new firmware).

If the version matches **CE00A474**, the update was successful.

13.5 Manual Installation Procedure on NVOS

13.5.1 Prerequisites

- SSH access to the system

13.5.2 Steps for Upgrade

1. (Optional) Check the current firmware version to see if the firmware version is different than CE00A474:

```
admin@nvos:~$ nv show platform firmware SSD
```

Example output:

```
                  operational
part-number       Virtium VTPM24CEXI080-BM110006
actual-firmware   CE00A400
fw-source         N/A
```

2. Fetch and install SSD firmware package:

```
admin@nvos:~$ nv action fetch platform firmware SSD scp://username[:password]@hostname/path/ssd_fw.pkg
```

```
admin@nvos:~$ nv action install platform firmware SSD files ssd_fw.pkg
```

- Upgrading or downgrading to SSD firmware version CE00A474 or later is supported on-the-fly; no power cycle is required or performed automatically.
- Upgrading or downgrading to SSD firmware version CE00A400 or CE00A450 will automatically trigger a system power cycle after the firmware update.

13.5.3 Expected Behavior After Installation

- The system will automatically perform a power cycle/cold reboot (if needed) after the firmware update.
- Once the system is back online, the SSD should operate normally.
- No additional steps are required from the customer after the update completes.

13.5.4 Verification

After the firmware update and system power cycle/cold reboot (if needed):

1. Log in to the system.
2. Run the following command to check the SSD firmware version:

```
admin@nvos:~$ nv show platform firmware SSD
```

Expected output:

```
                   operational
part-number        Virtium VTPM24CEXI080-BM110006
actual-firmware    CE00A474
fw-source          N/A
```

`actual-firmware` should show CE00A474 (indicating the new firmware).

If the version matches CE00A474, the update was successful.

14 Document Revision History

25.03.1003

Added:

- The command nv config lookup
- The ability to define a transceiver range in the command nv action install platform transceiver firmware files
- Note in the Transceiver Firmware Upgrade section.
- The field `supported-cable-ib-speed` to the output of the command nv show platform transceiver
- The section Plane-Port Telemetry
- The subsections “Subscription Paths” and “Minimum Sampling Intervals” to the section gNMI Streaming
- The command nv show system api versions

Updated:

- The subsection ZTP Configuration provisioning-script

Version 25.02.8008—April 2026

Added:

- The command nv show system reboot counters
- Commands in the following subsection gNMI Subscriptions
- The command nv show system gnmi-server status
- The command nv show system gnmi-server status client
- The command nv clear system gnmi-server status
- Commands in the following subsection NVUE Configuration Verify:
- nv config verify
- nv config verify revision
- nv config verify filename
- Commands in the following subsection gRPC Tunnel (gNMI Dial-out):
- nv show system grpc-tunnel
- nv show system grpc-tunnel server
- nv show system grpc-tunnel server
- nv show system grpc-tunnel server status
- nv show system grpc-tunnel server status connection
- nv unset system grpc-tunnel
- nv unset system grpc-tunnel server
- nv set/unset system grpc-tunnel server
- nv set/unset system grpc-tunnel server address

        - nv set/unset system grpc-tunnel server port
        - nv set/unset system grpc-tunnel server target-name
        - nv set/unset system grpc-tunnel server target-type
        - nv set/unset system grpc-tunnel server state
        - nv set/unset system grpc-tunnel server retry-interval
        - nv set/unset system grpc-tunnel server certificate
        - nv set/unset system grpc-tunnel server ca-certificate
    - IB Rounter Commands:
        - nv show ib router ib-subnet
        - nv show ib router ib-subnet subnet-name
        - nv show ib router ib-subnet subnet-name counters
        - nv show ib router ib-subnet subnet-name counters link
        - nv show ib router ib-subnet subnet-name gid
        - nv show ib router counters
        - nv action clear ib router counters
    - nvidia-ib-router to the subsection NVIDIA Models
    - The command nv show ib device ib-subnet
    - Transceiver failure event
    - Link diagnostic errors 1055, 1056, and 1057

Updated:

- The section (25.03.0600-VR-0.6) IB Router
- Output of the command nv show ib device
- Output of the command nv action change system profile
- Output of the command nv set/unset interface link ib-subnet
- Output of the command nv show system reboot reason
- Output of the command nv show system reboot history
- Note on the command nv set system ztp config-save

Version 25.02.7002—February 2026

Added

- Transceiver-Related Events in Event Management
- The sections IB Router and IB Router Commands
- The sections SSH for Remote Access and SSH for Remote Access Commands
- The command nv show system health component
- The command nv show system health component id
- The command nv action clear system health component
- The subsection gNMI Connection and Rate Limiting
- Note under section Transceiver Firmware Upgrade

Updated:

- Note under the section ZTP Configuration commands-list

Interface Counters

Added:

- The command nv show interface counters
- The command nv show interface counters link
- The command nv show interface counters ib
- The command nv show interface counters ib errors
- The command nv show interface counters ib drops
- The command nv show interface counters ib fast-recovery
- The command nv show interface counters nvl
- The command nv show interface counters nvl errors
- The command nv show interface counters nvl drops

Removed:

- The command nv show interface link

Access Control List (ACL's)

Added:

- The command nv set/unset system control-plane acl
- The command nv show system control-plane acl
- The command nv show system control-plane
- The command nv show system control-plane acl id
- The command nv show system control-plane acl statistics
- The command nv show system control-plane acl id statistics
- The command nv show system control-plane acl inbound
- The command nv show system control-plane acl outbound
- The command nv set/unset acl rule action recent

Uptated:

- The output of the command nv show acl
- The output of the command nv show interface acl
- The output of the command nv show interface acl id
- The output of the command nv show interface acl statistics
- The output of the command nv show interface acl statistics id
- The output of the command nv show interface acl inbound
- The output of the command nv show interface acl inbound control-plane
- The output of the command nv show interface acl outbound
- The output of the command nv show interface acl outbound control-plane
- The syntax of the command nv set/unset interface acl inbound control-plane

Interface/IP

Added:

- The command nv show interface ipv4 dhcp-client lease
- The command nv show interface ipv6 dhcp-client lease

Updated:

- Output of the command nv show interface

Replaced:

- The command `nv show interface ip` with nv show interface ipv4 and nv show interface ipv6
- The command `nv set/unset interface ip address` with nv set/unset interface ipv4 address and nv set/unset interface ipv6 address
- The command `nv set interface ip gateway` with nv set/unset interface ipv4 gateway and nv set/unset interface ipv6 gateway
- The command `nv set interface ip autoconf` with nv set interface ipv6 autoconf
- The command `nv set/unset interface ip arp-timeout` with nv set/unset interface ipv4 arp-timeout
- The command `nv set interface ip vrf` with nv set/unset interface vrf
- The command `nv set/unset interface ip dhcp-client state` with nv set/unset interface ipv4 dhcp-client state
- The command `nv set/unset interface ip dhcp-client set-hostname` with nv set/unset interface ipv4 dhcp-client set-hostname
- The command `nv set interface ip dhcp-client6 state` with nv set/unset interface ipv6 dhcp-client state
- The command `nv action renew interface ip dhcp-client` with nv action renew interface ipv4 dhcp-client
- The command `nv action renew interface ip dhcp-client6` with nv action renew interface ipv6 dhcp-client

Interface/Link

Added:

- The command nv show interface link phy health
- The command nv show interface link phy health histogram
- The command nv show interface link phy detail

Updated

- The `<auto>` parameter to be `enabled | disabled` instead of `on | off` in the command nv set interface link auto-negotiate
- Output of the command nv show interface

Version

25.02.6007

—October 2025

Added:

- `--expand` option to the command nv config show
- `--expand` option to the command nv config diff
- `--expand` option to the command nv config find
- Support for text file containing NVUE commands via nv config patch and nv config replace

Updated:

- The default of the command nv set system security password-hardening state to `disabled`
- The defalt of BIOS auto-update to `disabled`

Version

25.02.5030

—October 2025

Added:

- The section (25.03.0600-VR-0.6) Authorization Commands
- A note in the section (25.03.0600-VR-0.6) Remote Logging

Version

25.02.5019

/

25.02.5021

—September 2025

There are no changes to this version of the user manual. For further information on bug fixes and improvements, see the release notes of this software version.

Version

25.02.5002

—July 2025

Added:

- Two new operation codes (44 and 45) to the table Link Down Reason Indication
- New events to "System-Services-Related Events" section to the table Detailed Table of Events
- New event to "ASIC-Related Events" in the table Detailed Table of Events
- An update to dhcp-client-identifier to include in the Zero-Touch Provisioning section
- The command nv set/unset system ntp listen
- The command nv set/unset system cli inactive-timeout
- The command nv show system cli
- The command nv show system syslog selector
- The command nv show system syslog selector id
- The command nv show system syslog selector filter
- The command nv show system syslog selector filter id
- The command nv show system syslog selector rate-limit
- The command nv show system version image
- The command nv show system version packages installed
- The command nv show system version packages installed id
- The command nv show system ntp server detail (used to be `nv show system ntp serverquery`)

- The command nv show system date-time

Updated:

- The modes of the command nv action reboot system
- The output of the command nv show system serial-console
- The command "nv show system events last" to nv show system events --last
- The command "nv show system events recent" to nv show system events --recent
- The output of the command nv show system version
- The command `nv set system ssh-server inactivity-timeout` to nv set/unset system ssh-server inactive-timeout
- The command `nv set system ssh-serverports` to nv set/unset system ssh-server port
- The output of the command nv show system
- The output of the command nv show platform
- The output of the command nv show system syslog
- The output of the command nv show system syslog server
- The output of the command nv show system syslog server id
- The output of the command nv show system ntp
- The output of the command nv show system ntp server
- The command `nv set system timezone` to nv set/unset system date-time timezone
- The output of the command nv show system events to reflect Etc/UTC event timestamp

Removed:

- The command nv set/unset system syslog trap
- The command nv set system syslog format welf (use nv set system syslog format welf firewall-name <name>, instead)
-
- The "detail" option to the command nv show platform transceiver
- Note in Certificates Management section
- Note under Upgrading OS Software section

Version

25.02.4002

—May 2025

Added:

- The command nv show system aaa user spiffe-id
- The command nv set/unset system aaa user spiffe-id
- The command nv show system gnmi-server mtls
- The command nv set system gnmi-server mtls ca-certificate
- The table Link Down Reason Indication
- The command nv set acl rule action set dscp
- The subsection Set DSCP on Transit Traffic in the (25.03.0600-VR-0.6) Access Control List Configuration section

Updated:

- NVOS group IDs in the LDAP section
- The output of the command nv show system
- The output of the command nv show system cpu
- The remote-url options to include HTTPS in the following upload commands:
  - nv action upload system stats files
  - nv action upload system image files
  - nv action upload platform firmware files
  - nv action upload system documentation files
  - nv action upload system security tpm
  - nv action upload system config files
  - nv action upload system tech-support files
  - nv action upload ib ibdiagnet files
- The subsection Recovery Flow After SSD Wipe
- GRUB timeout style

Version

25.02.3000

—March 2025

Added:

- A subsection in the Troubleshooting section
- A note in the command nv action reset system factory-default
- Added commands to the (25.03.0600-VR-0.6) Role Based Access Control Commands section
- The command nv show system disk
- The command nv show system disk usage
- The command nv action erase system disk
- The command nv show system log file list
- The command nv show system security tpm oiak
- The command nv action import system security tpm oiak
- The command nv action delete system security tpm oiak
- The command nv show system aaa tacacs authentication
- The subsection ZTP Config provisioning-script
- The subsection Provisioning Script Example

Updated:

- The section (25.03.0600-VR-0.6) Role Based Access Control
- The Supported Events table
- Output of the command nv show platform transceiver
- Output of the command nv show system image
- Output of the command nv show system log to display log configurations
- Output of the command nv show system log file to display the contents of the latest syslog

- Output of the command nv show system ntp
- Output of the command nv show system aaa ldap
- The command `nv show system log files` to nv show system log file
- The "nvued" component to "nvue" (see (25.03.0600-VR-0.6) Logging Commands section)
- The command `nv show system aaa tacacs hostname` to nv show system aaa tacacs server
- The command `nv set system aaa tacacs hostname` to nv set system aaa tacacs server
- The command `nv set system aaa tacacs hostname auth-port` to nv set system aaa tacacs server auth-port
- The command `nv set system aaa tacacs hostname auth-mode` to nv set system aaa tacacs server auth-mode
- The command `nv set system aaa tacacs hostname secret` to nv set system aaa tacacs server secret
- The command `nv set system aaa tacacs hostname priority` to nv set system aaa tacacs server priority
- The command `nv set system aaa tacacs hostname timeout` to nv set system aaa tacacs server timeout
- The command `nv set system aaa tacacs auth-type` to nv set system aaa tacacs authentication mode
- The command `nv set system aaa tacacs hostname auth-type` to nv set system aaa tacacs server auth-mode
- The command `nv show system aaa radius hostname` to nv show system aaa radius server
- The command `nv show system aaa radius hostname id` to nv show system aaa radius server id
- The command `nv set system aaa radius hostname` to nv set system aaa radius server
- The command `nv set system aaa radius hostname auth-port` to nv set system aaa radius server auth-port
- The command `nv set system aaa radius hostname auth-type` to nv set system aaa radius server auth-type
- The command `nv set system aaa radius hostname secret` to nv set system aaa radius server secret
- The command `nv set system aaa radius hostname priority` to nv set system aaa radius server priority
- The command `nv set system aaa radius hostname retransmit` to nv set system aaa radius server retransmit
- The command `nv set system aaa radius hostname timeout` to nv set system aaa radius server timeout
- The command `nv show system aaa ldap hostname` to nv show system aaa ldap server
- The command `nv set system aaa ldap hostname id` to nv set system aaa ldap server id
- The command `nv set system aaa ldap hostname` to nv set system aaa ldap server

Removed:

- The command `nv set system aaa ldap ssl ca-list`

Version

25.02.2002

—January 2025

This is the initial release of NVOS software documentation for InfiniBand switches.

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. Neither NVIDIA Corporation nor any of its direct or indirect subsidiaries and affiliates (collectively: “NVIDIA”) make any representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice. Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer (“Terms of Sale”). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer’s own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer’s sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer’s product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA’s aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

NVIDIA