Lenovo Network

REST API Programming Guide

For Lenovo Cloud Network Operating System 10.8

Lenovo™

Note: Before using this information and the product it supports, read the general information in the *Safety information and Environmental Notices* and *User Guide* documents on the Lenovo *Documentation* CD and the *Warranty Information* document that comes with the product.

Second Edition (December 2018)

Contents

Preface

The *REST API Programming Guide for Lenovo CNOS 10.8* describes how to configure and use the Cloud Network Operating System 10.8 software on the following Lenovo RackSwitches:

- Lenovo RackSwitch G8272. For documentation on installing the switch physically, see the *Lenovo RackSwitch G8272 Installation Guide*.
- Lenovo RackSwitch G8296. For documentation on installing the switch physically, see the *Lenovo RackSwitch G8296 Installation Guide*.
- Lenovo RackSwitch G8332. For documentation on installing the switch physically, see the *Lenovo RackSwitch G8332 Installation Guide*.
- Lenovo ThinkSystem NE1032 RackSwitch. For documentation on installing the switch physically, see the *Lenovo ThinkSystem NE1032 RackSwitch Installation Guide*.
- Lenovo ThinkSystem NE1032T RackSwitch. For documentation on installing the switch physically, see the *Lenovo ThinkSystem NE1032T RackSwitch Installation Guide*.
- Lenovo ThinkSystem NE1072T RackSwitch. For documentation on installing the switch physically, see the *Lenovo ThinkSystem NE1072T RackSwitch Installation Guide*.
- Lenovo ThinkSystem NE10032 RackSwitch. For documentation on installing the switch physically, see the *Lenovo ThinkSystem NE10032 RackSwitch Installation Guide*.
- Lenovo ThinkSystem NE2572 RackSwitch. For documentation on installing the switch physically, see the *Lenovo ThinkSystem NE2572 RackSwitch Installation Guide*.
- Lenovo ThinkSystem NE0152T RackSwitch. For documentation on installing the switch physically, see the *Lenovo ThinkSystem NE0152T RackSwitch Installation Guide*.

Who Should Use This Guide

This guide is intended for network installers and system administrators engaged in configuring and maintaining a network. The administrator should be familiar with Ethernet concepts, IP addressing, Spanning Tree Protocol, and SNMP configuration parameters.

What You'll Find in This Guide

This guide will help you plan, implement, and administer Cloud NOS software. Where possible, each section provides feature overviews, usage examples, and configuration instructions. The following material is included:

This book contains the following chapters:

- Chapter 1, "Introduction," gives an overview of the Lenovo REST API and how to start the server.
- Chapter 2, "REST Server JSON Calls," describes the URIs and functions available in the REST API.
- Appendix A, "Getting Help and Technical Assistance," describes where to get help with your product.
- Appendix B, "Notices," contains legal notices.

Additional References

Additional information about installing and configuring the switch is available in the following guides:

- *Lenovo Network Application Guide for Lenovo Cloud Network Operating System 10.8*
- *Lenovo Network Command Reference for Lenovo Cloud Network Operating System 10.8*
- *Lenovo Network Release Notes for Lenovo Cloud network Operating System 10.8*
- *Lenovo Python Programming Guide for Lenovo Cloud Network Operating System 10.8*

Typographic Conventions

The following table describes the typographic styles used in this book.

Table 1. *Typographic Conventions*

Typeface or Symbol	Meaning	Example
`ABC123`	This type is used for names of commands, files, and directories used within the text. It also depicts on-screen computer output and prompts.	View the `readme.txt` file. `Switch#`
`ABC123`	This bold type appears in command examples. It shows text that must be typed in exactly as shown.	`Switch#` **`ping`**
<ABC123>	This italicized type appears in command examples as a parameter placeholder. Replace the indicated text with the appropriate real name or value when using the command. Do not type the brackets. This also shows book titles, special terms, or words to be emphasized.	To establish a Telnet session, enter: `Switch#` **`telnet`** *<IP address>* Read your *User's Guide* thoroughly.
`{}`	Command items shown inside brackets are mandatory and cannot be excluded. Do not type the brackets.	`Switch#` **`cp {ftp|sftp}`**
`[]`	Command items shown inside brackets are optional and can be used or excluded as the situation demands. Do not type the brackets.	`Switch#` **`configure [device]`**
`|`	The vertical bar (`|`) is used in command examples to separate choices where multiple options exist. Select only one of the listed options. Do not type the vertical bar.	`Switch#` **`cp {ftp|sftp}`**
<AaBb123>	This block type depicts menus, buttons, and other controls that appear in graphical interfaces.	Click the **<Save>** button.

Chapter 1. Introduction

The Lenovo REST Application Programming Interface (API) enables you to remotely configure and manage a Lenovo switch using REST, HyperText Transfer Protocol (HTTP), and Hyper Text Transfer Protocol over SSL (HTTPS).

The REST (REpresentational State Transfer) architecture has six constraints:

- Uniform Interface
- Stateless
- Cacheable
- Client-Server
- Layered Systems
- Code on Demand

The REST API is a JavaScript Object Notation-based (JSON) wrapper around Lenovo's Python On-Box Scripting interface. It is a component of Configuration, Management, and Reporting (CMR) on CNOS.

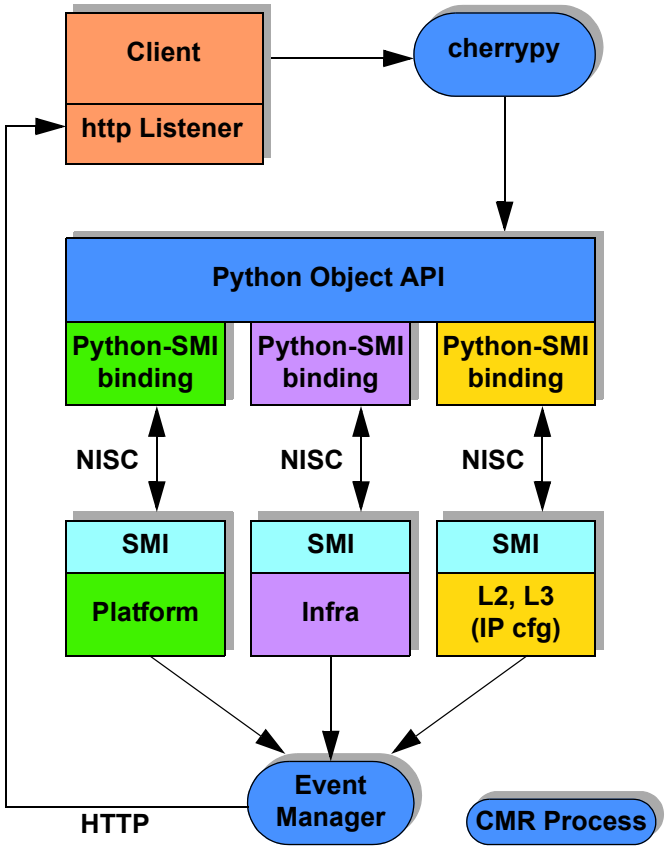
Note: The Lenovo REST API calls have been tested with:

- The Advanced Rest Client extension (version 6.19.17.118 or earlier) in Chrome
- The RESTClient extension in Firefox
- The Python3 http.client module

REST API Components

The following figure shows components of the REST API and JSON:

Figure 1. REST/JSON Components



The cherrypy server interprets the REST JSON code. When the cherrypy server receives a REST API request, it executes the appropriate Python code on Cloud NOS and translates it into a series of Simple Management Interface (SMI) calls. For each CLI connection through the console, SSH, or Telnet, a separate Cloud NOS process is spawned to service CLI commands.

Using the REST Server

This section discusses starting, stopping, and communicating with the REST server.

Starting and Stopping the REST Server

Use the CNOS CLI to start or stop the REST server.

Starting the REST Server

To start the REST server listening on the HTTPS port (443), in Global Configuration Mode on the switch, enter:

```
Switch(config)# feature restApi
```

To start the REST server listening in HTTP mode on port 8090, in Global Configuration Mode on the switch, enter:

```
Switch(config)# feature restApi http
```

This starts the REST server (cherrypy) listening on the specified port (443 or 8090) and writes the Process ID to the following PID file:

`/var/run/restfib`<*VRF ID*>`.pid`

where:

- *VRF ID* = `0` for the default Virtual Routing and Forwarding (VRF) ID
- *VRF ID* = `1` for the management Virtual Routing and Forwarding (VRF) ID

A separate REST server instance is created for each VRF ID created (one default, one management).

Stopping the REST Server

To stop the REST server, in Global Configuration Mode on the switch, enter:

```
Switch(config)# no feature restApi
```

This stops the REST server from listening on all ports for all VRF IDs.

Communicating with the REST Server

To log onto the REST server, use the URL:

`http://`*<management switch IP address>*`:`*<port>*`/nos/api/login`

The default *port* is 443.

Confirm adding a security exception. Enter your username and password.

Note: You must be a "`network-admin`" user to use the REST API. Requests from users with other roles will be rejected.

The REST API uses the following types of HTTP methods:

- `POST`
- `GET`
- `PUT`
- `DELETE`

Request Formats

The format of a URI or URL for a resource depends upon which type of request is being sent.

Table 2. *REST API URI/URL Conventions*

Request Type	URI Format
`POST`	`http://`*<switch address>*`:`*<port>*`/nos/api/cfg/`*<resource>* `[parameters={`*<parameters>*`}]`
`GET`	`http://`*<switch address>*`:`*<port>*`/nos/api/cfg/`*<resource>*`/`*<ID>* `http://`*<switch address>*`:`*<port>*`/nos/api/info/`*<resource>*`/`*<ID>*
`PUT`	`http://`*<switch address>*`:`*<port>*`/nos/api/cfg/`*<resource>* `[parameters={`*<parameters>*`}]`
`DELETE`	`http://`*<switch address>*`:`*<port>*`/nos/api/cfg/`*<resource>*`/`*<ID>*

where:

Parameter	Description
switch address:port	The switch IP address and port where the REST server is installed.
resource	Any network or switch resource, such as an interface or a VLAN.
parameters	Additional parameters related to the request, presented in JSON format.

The following example shows a PUT request for interface `ethernet1/1`:

```
PUT /nos/api/cfg/interface/Ethernet1%2F1
{
   "duplex": "full",
   "mtu": 1500,
   "admin_state": "up",
}
```

Note: When a port or other parameter in the URI has a slash (`/`) in it, such as `ethernet1/1`, you need to substitute the hexadecimal code for the slash (`%2F`, as in `ethernet1%2F1`) so the slash is not read as a directory delimiter. Slash characters *are* allowed in the JSON Request.

The following example shows the response to the previous PUT request:

```
PUT /nos/api/cfg/interface/Ethernet1%2F1
{
   "duplex": "full",
   "if_name": "Ethernet1/1",
   "mtu": 1500,
   "admin_state": "up",
   "mac_addr": "a897.dcf8.1101",
   "ifindex": "9",
   "oper_state": "up",
   "speed": "10000"
}
```

Getting the REST Server Status

To get the current status of the REST server, including the listening port number, from the CLI, enter:

```
Switch# display restApi server
```

Authenticating Users on the REST Server

To log onto the switch via the REST server, use the following URL:

`https://`*<IP address>*`/nos/api/login`

where *IP address* is the management IP address of the switch you are accessing.

Your user session expires based on the switch timeout value, which defaults to 10 minutes.

To log out of the switch via the REST server, use the following URL:

`https://`*<IP address>*`/nos/api/logout`

Server Security

The REST API uses the local user database in CNOS on the switch for authentication. All REST requests must be issued by a "`network-admin`" user. Requests made by any other type of user will be rejected by the REST API server.

The REST server uses cookies to identify sessions. Specifically, a cookie is assigned for each session, and its passback will be requested by the REST server. A REST API client must first issue a "Set-Cookie" request and then must pass the cookie back on all subsequent REST requests.

HTTPS Support

When REST API via HTTPS is enabled by default, a self-signed certificate is generated automatically.

Note: Lenovo recommends using CSR or CA signed certificates rather than self-signed certificate. For more information on how to generate CSR or CA certificates, see the *CNOS Application Guide*.

To refresh a self-signed certificate, use the following steps:

1. Disable the REST server:

```
Switch(config)# no feature restApi
```

To verify that the REST server is not running, enter:

```
Switch(config)# display restApi server
rest server disabled port: 8090(HTTP)
```

2. Enter the Public Key Infrastructure (PKI) configuration mode.

```
Switch(config)# pki rest_mgmt
Switch(config-pki)#
```

3. Create the certificate:

```
Switch(config-pki)#host-cert generate
Country Name (2 letter code) [US]:
State or Province Name (full name) [California]:
Locality Name (eg, city) [Santa Clara]:
Organization Name (eg, company) [Lenovo Networking Operating System]:
Organizational Unit Name (eg, section) [Network Engineering]:
Common Name (eg, FQDN or YOUR name) []: netuser
Email (eg, email address) []: netuser@lenovo.com
Confirm generate certificate? (y/n) [n] y
.......+++
.........+++
Host certificate generation succeeded
```

Note: The default values are in square brackets ([text]); press **Enter** to use the default values.

4. Re-enable the REST server:

```
Switch(config)# feature restApi
```

5. Make sure the REST server is running:

```
Switch(config)# display restApi server
rest server enabled port: 443
restApi pki rest_mgmt vrf management
restApi pki rest_default vrf default
```

6. The REST PKI profiles are automatically generated.

To display host certificate information, use the following commands: :

```
Switch(config)#display pki rest_mgmt host-certificate
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 0 (0x0)
    Signature Algorithm: sha512WithRSAEncryption
        Issuer: C=US, ST=California, L=Santa Clara, O=Lenovo Networking
Operating System, OU=Network Engineering,
CN=acomsa/emailAddress=netuser@lenovo.com
        Validity
            Not Before: May  3 14:49:49 2017 GMT
            Not After : May  3 14:49:49 2018 GMT
        Subject: C=US, ST=California, L=Santa Clara, O=Lenovo Networking
Operating System, OU=Network Engineering,
CN=acomsa/emailAddress=netuser@lenovo.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:d2:e6:5d:11:c1:0c:f0:5e:75:09:ac:ab:77:2b:
                    a2:c2:ca:fd:33:79:f9:58:6c:c6:d9:89:87:a4:d8:
                    94:79:ab:ca:f2:15:f3:ab:43:66:27:2f:8f:40:76:
                    7f:ed:4c:5a:e2:23:18:98:68:fe:4b:51:bf:4a:6b:
                    64:08:4f:00:90:0e:df:71:d7:c4:db:48:99:4f:3d:
                    47:4b:ae:0a:9a:ba:d8:f0:15:93:4e:c0:6d:2c:64:
                    a9:1f:c0:a7:6f:7f:4f:87:2d:b5:c7:8a:d5:09:37:
                    5c:8b:6f:14:b5:e7:8c:5d:99:da:ae:20:2c:0d:94:
                    b3:c3:f8:4c:5f:04:8f:71:4f:19:b2:18:11:64:e4:
                    9a:96:41:2b:bf:de:9a:87:32:6b:a5:22:f3:eb:32:
                    da:c5:ac:c8:d4:cf:83:14:6a:39:23:b9:49:2e:bc:
                    ec:84:e6:5c:f9:d6:df:2d:97:e7:f3:dd:cb:6d:c0:
                    94:e1:a1:9a:94:ea:3a:65:04:e7:63:45:fa:70:7d:
                    f6:89:2d:af:7d:bf:d4:7d:f2:f1:45:b7:a4:11:16:
                    29:c4:4a:56:58:63:6e:b6:4d:6a:aa:c8:2e:c0:7b:
                    15:b5:7b:bf:00:00:f6:9c:75:6a:cd:50:2d:6e:68:
                    24:74:77:dc:29:dc:7e:35:b0:4a:02:f9:76:b0:7c:
                    65:23
```

```
  Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Basic Constraints: critical
                CA:TRUE
            X509v3 Key Usage: critical
                Certificate Sign, CRL Sign
            X509v3 Subject Key Identifier:

51:7A:5E:95:9D:0E:23:17:57:DF:13:63:D1:07:A6:05:07:B3:38:7F
Signature Algorithm: sha512WithRSAEncryption
         74:b4:16:bf:06:a9:69:8f:dc:8f:de:cf:5d:18:f8:ba:82:71:
         b4:8b:8c:22:b4:1e:66:55:d3:3f:a1:71:cc:7b:1a:bd:fd:5b:
         56:d7:c8:4c:4c:32:09:47:1c:7e:8a:f1:f6:f4:67:95:d6:88:
         7f:f5:ad:af:09:e8:5c:ca:46:54:93:71:38:b6:00:e8:b3:fa:
         cc:71:e7:cb:67:ac:8f:ec:22:01:3e:da:54:04:f8:77:3d:2c:
         78:80:a1:01:6e:d6:19:23:1a:f2:d0:8e:af:71:e3:1a:b0:a5:
         9b:fa:53:04:eb:92:2b:b0:b5:c2:51:d0:e0:85:b5:04:f7:24:
         5b:20:58:76:f8:e3:bc:a6:c4:15:2a:5a:ee:60:bb:eb:f5:96:
         ce:2d:9a:78:bd:5b:c5:68:a3:c7:5a:41:a4:48:43:5d:f6:8a:
         ee:9f:cf:e8:8c:48:b6:2a:9a:93:aa:ed:00:87:2b:12:92:b6:
         2f:1d:9b:70:43:57:98:a2:70:16:8e:0c:7d:ac:b2:9e:d0:99:
         2d:76:2f:20:f7:49:c9:ac:08:e2:cc:a6:4e:10:12:bd:c0:15:
         a3:e6:1c:6e:5d:96:8f:31:ab:19:92:42:70:e7:c0:3a:f7:cb:
         43:a0:c4:db:99:68:37:ca:69:e7:e0:35:52:7d:6c:ec:9e:0a:
         56:25:4b:09

Switch(config)#display pki rest_default host-certificate
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:
            95:22:0b:f1:2b:b8:96:69
    Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, ST=California, L=Santa Clara, O=Lenovo Network
Operating System CNOS, OU=Network Engineering, CN=0.0.0.0
        Validity
            Not Before: Mar  7 12:53:02 2017 GMT
            Not After : Mar  7 12:53:02 2018 GMT
        Subject: C=US, ST=California, L=Santa Clara, O=Lenovo Network
Operating System CNOS, OU=Network Engineering, CN=0.0.0.0
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
                    00:e3:81:8f:dd:a2:d9:ef:9b:3e:50:4f:f6:79:e2:
                    d2:07:06:3e:db:46:fd:05:7b:ea:84:f0:34:a1:b7:
                    e7:4c:f0:3d:c3:b0:c0:82:1d:60:85:b5:ec:82:ea:
                    e2:65:a3:a3:6b:27:f5:17:b1:fe:52:c1:ea:4c:40:
                    55:0a:c0:2f:6f:4c:42:ef:74:72:ef:a4:5b:b2:4d:
                    90:74:97:48:51:bd:d8:9b:56:2c:ee:e4:41:5e:4f:
                    b9:0a:31:91:c6:08:94:cb:21:6e:d2:69:0d:db:12:
                    56:2a:33:2c:1b:de:53:93:2d:f4:00:74:38:65:e3:
                    f5:2d:09:f3:14:36:63:23:33:d8:9d:1b:d6:ba:4c:
                    8f:0c:de:e7:3e:56:d4:4e:ab:3c:cc:27:a1:0f:15:
                    e5:8c:a8:f0:cf:84:7c:3f:3d:23:19:71:25:7d:19:
                    26:b6:79:47:a1:f6:6c:ee:91:2f:db:55:3e:17:7a:
                    89:ab:43:6e:73:9b:bc:b7:54:b6:83:d7:a5:9a:5c:
                    8f:d0:a6:d1:65:f0:d2:6a:70:25:ce:9b:9a:06:49:
                    4e:5a:cd:d5:4c:96:1f:84:f1:b9:97:ea:a9:de:c5:
                    26:80:ee:48:3b:aa:b8:4c:fd:bc:71:0e:96:40:64:
                    38:20:da:0e:a4:42:a9:95:ae:43:de:14:2b:2a:4c:
                    3e:a9
```

```
                Exponent: 65537 (0x10001)
        X509v3 extensions:
            X509v3 Subject Key Identifier:

D5:FC:6B:30:CD:D9:7B:4D:57:30:80:6A:AD:96:E6:02:27:06:EF:DA
            X509v3 Authority Key Identifier:

keyid:D5:FC:6B:30:CD:D9:7B:4D:57:30:80:6A:AD:96:E6:02:27:06:EF:DA
            X509v3 Basic Constraints:
                CA:FALSE
    Signature Algorithm: sha256WithRSAEncryption
         b1:e5:ad:cb:9c:c9:fe:7a:8f:2f:73:2a:eb:76:cc:9d:f2:41:
         16:b7:c6:5b:aa:84:30:37:b2:8c:f3:5a:71:2e:77:28:56:1c:
         42:76:6c:fa:8c:ef:53:4d:db:34:3d:1c:45:c1:80:64:1c:04:
         18:8e:79:8b:d7:92:55:13:89:ad:d4:d0:47:e0:d4:10:db:37:
         72:5d:a2:45:f8:7d:ed:fd:18:f7:04:c8:64:98:2d:c5:76:43:
         ef:1e:33:c8:05:63:10:cf:db:28:e5:8d:c1:6d:4b:2e:2a:54:
         df:c1:96:34:6f:3a:64:18:f3:97:7f:2a:58:6b:f2:8e:ee:10:
         da:48:1d:58:47:9d:5d:26:44:22:d6:10:ce:11:68:21:db:ea:
         e8:3f:1a:5c:d0:33:2b:92:23:f5:44:de:43:32:d6:b7:fc:ef:
         76:97:b7:65:b4:f2:f5:a9:d4:7e:1a:3d:fb:f3:ce:c0:2f:8a:
         fb:33:98:a4:5c:9a:44:9f:10:81:24:78:d5:36:7d:3c:b5:3c:
         da:2d:6c:7a:48:8b:a3:4c:0a:2b:99:1f:23:ef:1a:4e:3d:b7:
         ea:b2:41:dc:20:54:d2:06:6f:b9:10:7a:58:55:f3:ba:ba:72:
         23:f3:11:ec:32:11:71:4d:70:5a:2a:6c:07:c2:0d:75:25:aa:
         77:2d:f2:af
```

REST Server Limitations

The following limitations apply to the REST server:

- Authentication via RADIUS or TACACS+ is not supported.
- REST API calls can *only* be made by a "`network-admin`" user. Requests made by any other type of user will be rejected by the REST API server.
- The only MIME type supported is "`application/json`". Any other values, including no MIME type, will be rejected.
- Each request from a client must contain all information necessary for the REST server to fulfill the request. Requests with partial information will be ignored.

Chapter 2. REST Server JSON Calls

This chapter contains the JavaScript Object Notation (JSON) calls you can make to the REST server on the switch.

- "NOS Copy" on page 411 - lists URIs and functions with regards to copying NOS images and configuration files
- "NPA" on page 425 - lists URIs and functions with regards to Network Policy Agent (NPA)
- "NTP" on page 471 - lists URIs and functions with regards to Network Time Protocol (NTP)
- "OSPF" on page 479 - lists URIs and functions with regards to Open Shortest Path First (OSPF)
- "PKA" on page 543 - lists URIs and functions with regards to Public Key Authentication (PKA)
- "PKI" on page 551 - lists URIs and functions with regards to Private Key Infrastructure (PKI)
- "Private VLAN" on page 571 - lists URIs and functions with regards to private virtual LANs information and configuration
- "QoS over L3" on page 585 - lists URIs and functions with regards to QoS over L3 information and configuration
- "RADIUS" on page 587 - lists URIs and functions with regards to Remote Authentication Dial-In User Service (RADIUS)
- "REST" on page 603 - lists URIs and functions with regards to REST information and configuration
- "Route Maps" on page 607 - lists URIs and functions with regards to route maps information and configuration
- "Security Mode" on page 609 - lists URIs and functions with regards to the switch security mode
- "sFlow" on page 613 - lists URIs and functions with regards to sampled flow (sFlow)
- "SNMP" on page 621 - lists URIs and functions with regards to Simple Network Management Protocol (SNMP)
- "SSH" on page 631 - lists URIs and functions with regards to Secure Shell (SSH) management
- "Startup Information" on page 635 - lists URIs and functions with regards to startup information, such as the current startup image or Zero Touch Provisioning (ZTP) settings
- "STP" on page 641 - lists URIs and functions with regards to Spanning Tree Protocol (STP)
- "System" on page 653 - lists URIs and functions with regards to system properties
- "System Configuration" on page 655 - lists URIs and functions with regards to system configuration, such as hostname or switch clock settings
- "System Information" on page 673 - lists URIs and functions with regards to system information, such as the status of system resources
- "Telemetry" on page 685 - lists URIs and functions with regards to telemetry information and configuration

- "Telnet" on page 721 - lists URIs and functions with regards to telnet information and configuration
- "TACACS+" on page 725 - lists URIs and functions with regards to Terminal Access Controller Access-Control System Plus (TACACS+)
- "vLAG" on page 737 - lists URIs and functions with regards to Virtual Link Aggregation Group (vLAG)
- "VLAN" on page 753 - lists URIs and functions with regards to virtual LANs
- "VLAN Interface Properties" on page 765 - lists URIs and functions with regards to Switch Virtual Interfaces (SVIs)
- "VRF" on page 773 - lists URIs and functions with regards to Virtual Routing and Forwarding (VRF)
- "VRRP" on page 779 - lists URIs and functions with regards to Virtual Router Redundancy Protocol (VRRP)

AAA

The following Authentication, Authorization and Accounting (AAA) URIs are available:

- /nos/api/cfg/aaa/accounting/default GET, PUT
- /nos/api/cfg/aaa/authorization/commands/default GET, PUT
- /nos/api/cfg/aaa/authorization/config-commands/default GET, PUT
- /nos/api/cfg/aaa/authentication/login/console GET, PUT
- /nos/api/cfg/aaa/authentication/login/default GET, PUT
- /nos/api/cfg/aaa/authentication/login/error-enable GET, PUT
- /nos/api/cfg/aaa/local/authentication/ GET, PUT
- /nos/api/cfg/aaa/user/default-role GET, PUT
- /nos/api/info/aaa/groups GET

The following AAA commands are available:

- Get AAA Accounting Configuration
- Update AAA Accounting Configuration
- Get AAA User EXEC Commands Authorization Setting
- Update AAA User EXEC Commands Authorization Settings
- Get AAA Configuration Commands Authorization Settings
- Update AAA Configuration Commands Authorization Settings
- Get AAA Console User Login Authentication Configuration
- Update AAA Console User Login Authentication Configuration
- Get AAA Remote User Login Authentication Configuration
- Update AAA Remote User Login Authentication Configuration
- Get AAA Authentication Login Error Message Status
- Update AAA Authentication Login Error Message Status
- Get AAA Local Authentication Configuration
- Update AAA Local Authentication Configuration
- Get AAA User Default Role Status
- Update AAA User Default Role Status
- Get AAA Groups

Get AAA Accounting Configuration

Gets the AAA accounting configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/aaa/accounting/default
Request Body (JSON)	

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Update AAA Accounting Configuration

Updates the AAA accounting configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/aaa/accounting/default

Request Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Get AAA User EXEC Commands Authorization Setting

Gets the current User EXEC command mode authorization settings.

Request

Method Type	GET
Request URI	/nos/api/cfg/aaa/authorization/commands/default
Request Body (JSON)	

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Update AAA User EXEC Commands Authorization Settings

Updates the AAA User EXEC command mode authorization settings.

Request

Method Type	PUT
Request URI	/nos/api/cfg/aaa/authorization/commands/default

Request Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Get AAA Configuration Commands Authorization Settings

Gets the current configuration command mode authorization settings.

Request

Method Type	GET
Request URI	/nos/api/cfg/aaa/authorization/config-commands/default
Request Body (JSON)	

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Update AAA Configuration Commands Authorization Settings

Updates the AAA User EXEC command mode authorization settings.

Request

Method Type	PUT
Request URI	/nos/api/cfg/aaa/authorization/config-commands/default

Request Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`).

Get AAA Console User Login Authentication Configuration

Gets the current console user login authentication configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/aaa/authentication/login/console
Request Body (JSON)	

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local] [none]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`), `none`.

Update AAA Console User Login Authentication Configuration

Updates the console user login authentication configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/aaa/authentication/login/console

Request Body (JSON)	`{`
	`  "methods":"[group {`*group_name*`}+] [local] [none]"`
	`}`

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`), `none`.

Response

Response Body (JSON)	`{`
	`  "methods":"[group {`*group_name*`}+] [local] [none]"`
	`}`

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`), `none`.

Get AAA Remote User Login Authentication Configuration

Gets the current remote user login authentication configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/aaa/authentication/login/default
Request Body (JSON)	

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local] [none]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`), `none`.

Update AAA Remote User Login Authentication Configuration

Updates the remote user login authentication configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/aaa/authentication/login/default

Request Body (JSON)	{
	"methods":"[group {*group_name*}+] [local] [none]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`), `none`.

Response

Response Body (JSON)	{
	"methods":"[group {*group_name*}+] [local] [none]"
	}

where:

Element	Description
`methods`	The AAA accounting methods; one of `group` - followed by a list of maximum 8 AAA groups (optionally followed by `local`), `none`.

Get AAA Authentication Login Error Message Status

Checks if error messages are displayed when users fail to authenticate.

Request

Method Type	GET
Request URI	/nos/api/cfg/aaa/authentication/login/error-enable
Request Body (JSON)	

Response

Response Body (JSON)	{
	"status":"{enable|disable}"
	}

where:

Element	Description
`status`	The status of authentication login error messages; one of *enable*, *disable*.

Update AAA Authentication Login Error Message Status

Updates the status of authentication login error messages.

Request

Method Type	PUT
Request URI	/nos/api/cfg/aaa/authentication/login/error-enable

Request Body (JSON)	{
	"status":"{enable|disable}"
	}

where:

Element	Description
`status`	The status of authentication login error messages; one of *enable*, *disable*.

Response

Response Body (JSON)	{
	"status":"{enable|disable}"
	}

where:

Element	Description
`status`	The status of authentication login error messages; one of *enable*, *disable*.

Get AAA Local Authentication Configuration

Gets the AAA local authentication configuration, such as the maximum number of unsuccessful authentication attempts before a user is locked out.

Request

Method Type	GET
Request URI	/nos/api/cfg/aaa//local/authentication
Request Body (JSON)	

Response

Response Body (JSON)	{
	"maxfail_attempts":{*maxfail_attempts*}
	}

where:

Element	Description
`maxfail_attempts`	The maximum number of unsuccessful authentication attempts before a user is locked out; an integer from 1-25. Default value: 3.

Update AAA Local Authentication Configuration

Updates the AAA local authentication configuration, such as the maximum number of unsuccessful authentication attempts before a user is locked out.

Request

Method Type	PUT
Request URI	/nos/api/cfg/aaa/local/authentication
Request Body (JSON)	

where:

Element	Description
`maxfail_attempts`	The maximum number of unsuccessful authentication attempts before a user is locked out; an integer from 1-25. Default value: 3.

Response

Response Body (JSON)	{
	"maxfail_attempts":{*maxfail_attempts*}
	}

where:

Element	Description
`maxfail_attempts`	The maximum number of unsuccessful authentication attempts before a user is locked out; an integer from 1-25. Default value: 3.

Get AAA User Default Role Status

Checks if users are allowed to login even if the TACACS+ server does not provide a default role. The default role is network-operator.

Request

Method Type	GET
Request URI	/nos/api/cfg/aaa/user/default-role
Request Body (JSON)	

Response

Response Body (JSON)	{
	"status":"{enable|disable}"
	}

where:

Element	Description
`status`	The status of allowing users to login even if the TACACS+ server does not provide a role; one of *enable, disable*.

Update AAA User Default Role Status

Updates the status of allowing users to login even if the TACACS+ server does not provide a role. The default role is network-operator.

Request

Method Type	PUT
Request URI	/nos/api/cfg/aaa/user/default-role

Request Body (JSON)	{
	"status":"{enable|disable}"
	}

where:

Element	Description
`status`	The status of allowing users to login even if the TACACS+ server does not provide a role; one of *enable*, *disable*.

Response

Response Body (JSON)	{
	"status":"{enable|disable}"
	}

where:

Element	Description
`status`	The status of allowing users to login even if the TACACS+ server does not provide a role; one of *enable*, *disable*.

Get AAA Groups

Gets AAA group information.

Request

Method Type	GET
Request URI	/nos/api/info/aaa/groups
Request Body (JSON)	

Response

Response Body (JSON)	`[`
	`  {`
	`  "group_name": "{`*group_name*`}",`
	`  "type": "{TACACS+}"`
	`  }`
	`]`

where:

Element	Description
`group_name`	The name of the AAA group (string).
`type`	The type of the AAA group; for example *TACACS+*.

Note: Currently, the only supported AAA group type is *TACACS+*.

ARP

The following ARP URIs are available:

- /nos/api/cfg/arp — GET, PUT
- /nos/api/cfg/arp_interface — GET
- /nos/api/cfg/arp_interface/*<if_name>* — GET, PUT
- /nos/api/cfg/arp_entry — GET
- /nos/api/cfg/arp_entry/*<if_name>* — GET, POST
- /nos/api/cfg/arp_entry/*<if_name>*/*<ip_addr>* — GET, PUT, DELETE
- /nos/api/cfg/arp_refresh — GET, PUT

The following ARP commands are available:

- Get ARP System Properties
- Update ARP System Properties
- Get ARP Properties of All Interfaces
- Get ARP Interface Properties
- Update ARP Interface Properties
- Get Static ARP Entries of All Interfaces
- Get Static ARP Entries of One Interface
- Create Static ARP Entry
- Get Static ARP Entry
- Update Static ARP Entry
- Delete Static ARP Entry
- Get ARP Refresh Configuration
- Update ARP Refresh Configuration

Get ARP System Properties

Gets global ARP properties of the system.

Request

Method Type	GET
Request URI	/nos/api/cfg/arp
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  "ageout_time": "*<ageout_time>*"
	}

where:

Element	Description
`ageout_time`	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

Update ARP System Properties

Updates the global ARP properties of the system.

Request

Method Type	PUT
Request URI	/nos/api/cfg/arp

Request Body
 	
(JSON)
	{
	  "ageout_time": "*<ageout_time>*"
	}

where:

Element	Description
`ageout_time`	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

Response

Response Body
 	
(JSON)
	{
	  "ageout_time": "*<ageout_time>*"
	}

Get ARP Properties of All Interfaces

Gets ARP properties of all interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/arp_interface
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "if_name": "*<if_name>*",
	  "ageout_time": "*<ageout_time>*"
	  }
	]

where:

Element	Description
`if_name`	IP interface name (string). **Note:** The interface must exist.
`ageout_time`	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

Get ARP Interface Properties

Gets ARP properties of one interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/arp_interface/<*if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "if_name": "<*if_name*>",
	  "ageout_time": "<*ageout_time*>"
	  }
	]

where:

Element	Description
`if_name`	IP interface name (string). **Note:** The interface must exist.
`ageout_time`	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

Update ARP Interface Properties

Updates the ARP properties of one interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/arp_interface/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The IP interface name (string). **Note:** The interface must exist.
ageout_time	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "if_name": "`<*if_name*>`",`
	`  "ageout_time": "`<*ageout_time*>`"`
	`  }`
	`]`

where:

Element	Description
`if_name`	The IP interface name (string).
`ageout_time`	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

Get Static ARP Entries of All Interfaces

Gets all static ARP entries of all interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/arp_entry
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "if_name": "`*<if_name>*`",`
	`  "ip_addr": "`*<ip_addr>*`",`
	`  "mac_addr": "`*<mac_addr>*`"`
	`  }`
	`]`

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`ip_addr`	The IP address.
`mac_addr`	The MAC address in the following format: xxxx.xxxx.xxxx.

Get Static ARP Entries of One Interface

Gets all static ARP entries under the specified interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/arp_entry/<*if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "if_name": "<*if_name*>",
	  "ip_addr": "<*ip_addr*>",
	  "mac_addr": "<*mac_addr*>"
	  }
	]

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`ip_addr`	The IP address.
`mac_addr`	The MAC address in the following format: xxxx.xxxx.xxxx.

Create Static ARP Entry

Creates a static ARP entry under the specified interface.

Request

Method Type	POST
Request URI	/nos/api/cfg/arp_entry/<*if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"ip_addr": "<*ip_addr*>",
	"mac_addr": "<*mac_addr*>"
	}

where:

Element	Description
if_name	Interface name. **Note:** The interface must exist.
ip_addr	The IP address.
mac_addr	The MAC address in the following format: xxxx.xxxx.xxxx.

Response

Response Body
 	
(JSON)
	[
	{
	"if_name": "<*if_name*>",
	"ip_addr": "<*ip_addr*>",
	"mac_addr": "<*mac_addr*>"
	}
	]

Get Static ARP Entry

Gets one static ARP entry under the specified interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/arp_entry/<*if_name*>/<*ip_addr*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"ip_addr": "<*ip_addr*>",
	"mac_addr": "<*mac_addr*>"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`ip_addr`	The IP address.
`mac_addr`	The MAC address in the following format: xxxx.xxxx.xxxx.

Update Static ARP Entry

Updates properties of one static ARP entry under the specified interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/arp_entry/<*if_name*>/<*ip_addr*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"ip_addr": "<*ip_addr*>",
	"mac_addr": "<*mac_addr*>"
	}

where:

Element	Description
if_name	Interface name. **Note:** The interface must exist.
ip_addr	The IP address.
mac_addr	The MAC address in the following format: xxxx.xxxx.xxxx.

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"ip_addr": "<*ip_addr*>",
	"mac_addr": "<*mac_addr*>"
	}

Delete Static ARP Entry

Deletes a static ARP entry under the specified interface.

Note: If the specified *ip_addr* is `all`, all static ARP entries under specified interface will be deleted.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/arp_entry/<*if_name*>/<*ip_addr*>
Request Body (JSON)	

Get ARP Refresh Configuration

Gets the ARP refresh configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/arp_refresh
Request Body (JSON)	

Response

Response Body (JSON)	`{`
	`  "state":"{enabled|disabled}"`
	`}`

where:

Element	Description
`state`	The status of ARP refresh on the switchl one of *enabled, disabled*.

Update ARP Refresh Configuration

Updates the ARP refresh configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/arp_refresh

Request Body (JSON)	`{`
	`  "state":"{enabled|disabled}"`
	`}`

where:

Element	Description
`state`	The status of ARP refresh on the switchl one of *enabled*, *disabled*.

Response

Response Body (JSON)	`{`
	`  "state":"{enabled|disabled}"`
	`}`

where:

Element	Description
`state`	The status of ARP refresh on the switchl one of *enabled*, *disabled*.

BFD

The following Bidirectional Forwarding Detection URIs are available:

- /nos/api/info/bfd GET
- /nos/api/info/bfd/details GET
- /nos/api/info/bfd/application/<*protocol_name*> GET
- /nos/api/info/bfd/application_details/<*protocol_name*> GET
- /nos/api/info/bfd/loc_addr/<*loc_addr*> GET
- /nos/api/info/bfd/rem_addr/<*rem_addr*> GET
- /nos/api/cfg/bfd POST
- /nos/api/cfg/bfd/interface/<*if_name*> POST
- /nos/api/cfg/bfd/interface/<*if_name*>/neighbors POST
- /nos/api/cfg/bfd/interface/<*if_name*>/neighbors/<*loc_disc*> DELETE
- /nos/api/cfg/bfd/multihop_peer/<*rem_addr*> POST

The following BFD commands are available:

- Get all BFD Sessions
- Get all BFD Sessions Details
- Get BFD Sessions for the Specified Protocol
- Get all BFD Session Details for the Specified Protocol
- Get BFD Sessions with the Specified Local Address
- Get BFD Sessions with the Specified Remote Address
- Change BFD Global Options
- Change BFD Interface Options
- Create a New BFD Session
- Delete a BFD Session
- Set or Unset BFD Multihop Session Options

Get all BFD Sessions

Get information about all BFD sessions.

Request

Method Type	GET
Request URI	/nos/api/info/bfd
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	{
	"loc_addr": "*<loc_addr>*",
	"rem_addr": "*<rem_addr>*",
	"loc_disc": "*<loc_disc>*",
	"rem_disc": "*<rem_disc>*",
	"RH/RS": "*<RH/RS>*",
	"holdown": "*<holdown>*",
	"mult": "*<mult>*",
	"sess_state": "*<sess_state>*",
	"if_name": "*<if_name>*"
	}
	]

where:

Element	Description
`loc_addr`	BFD session source IP address (string); a valid IPv4 or IPv6 address.
`rem_addr`	BFD session destination IP address (string); a valid IPv4 or IPv6 address.
`loc_disc`	Unique number used by the local system to identify the BFD session; an integer from *1-2147483647*.
`rem_disc`	Unique number used by the remote system to identify the BFD session; an integer from *1-2147483647*.
`RH/RS`	Remote Heard/Remote State (string); one of *UP, DOWN, ADMIN_DOWN*.
`holdown`	If no BFD packet is received in the specified number of milliseconds, the session will be declared down; an integer from *150-2997*.
`mult`	The number of times a packet is missed before BFD declares the neighbor down; an integer from *3-50*.

Element	Description
`sess_state`	BFD session state (string); one of *UP, DOWN, ADMIN_DOWN*.
`if_name`	Interface on which the BFD session is active. **Note:** The interface must exist.

Get all BFD Sessions Details

Get detailed information about all BFD sessions.

Request

Method Type	GET
Request URI	/nos/api/info/bfd/details
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	{
	"loc_addr": "*<loc_addr>*",
	"rem_addr": "*<rem_addr>*",
	"loc_disc": *<loc_disc>*,
	"rem_disc": *<rem_disc>*,
	"RH/RS": "*<RH/RS>*",
	"holdown": *<holdown>*,
	"mult": *<mult>*,
	"sess_state": "*<sess_state>*",
	"if_name": "*<if_name>*"
	"echo_mode": "*<echo_mode>*",
	"gtsm": "*<gtsm>*",
	"gtsm_ttl": *<value>*,
	"minTxInt": *<minTxInt>*,
	"minRxInt": *<minRxInt>*,
	"negotiated minRxInt": *<negotiated minRxInt>*,
	"negotiated multiplier": *<negotiated multiplier>*,
	"rx Count": *<rx Count>*,
	"tx Count": *<tx Count>*,
	"registered protocols": "*<registered protocols>*",
	"uptime": "*<uptime>*"
	}
	]

where:

Element	Description
`loc_addr`	BFD session source address (string); a valid IPv4 or IPv6 address.
`rem_addr`	BFD session destination IP address (string); a valid IPv4 or IPv6 address.
`loc_disc`	Unique number used by the local system to identify the BFD session; an integer from *1-2147483647*.
`rem_disc`	Unique number used by the remote system to identify the BFD session; an integer from *1-2147483647*.

Element	Description
RH/RS	Remote Heard/Remote State (string); one of *UP*, *DOWN*, *ADMIN_DOWN*.
holdown	If no BFD packet is received in the specified number of milliseconds, the session will be declared down; an integer from *150-2997*.
mult	The number of times a packet is missed before BFD declares the neighbor down; an integer from *3-50*.
sess_state	BFD session state (string); one of *UP*, *DOWN*, *ADMIN_DOWN*.
if_name	Interface on which the BFD session is active. **Note:** The interface must exist.
echo_mode	Whether the BFD session has echo mode enabled (string); one of *enabled*, *disabled*.
gtsm	Whether the BFD session has Generalized TTL Security Mechanism (GTSM) enabled; one of *enabled*, *disabled*.
gtsm_ttl	Displays BFD GTSM Time-To Live (TTL) value; an integer from *1-255*.
minTxInt	Rate at which BFD control packets will be sent to BFD neighbors; an integer from *50-999*.
minRxInt	Specifies the rate at which BFD control packets will be expected to be received from BFD neighbors; an integer from *50-999*.
negotiated minRxInt	Specifies the negotiated rate at which BFD control packets will be received from BFD neighbors; an integer from *50-999*.
negotiated multiplier	The number of times a packet is missed before BFD declares the session down; an integer from *3-50*.
rx Count	The number of received BFD packets; a positive integer.
tx Count	The number of sent BFD packets; a positive integer.
registered protocols	Protocol for which the BFD session is active (string); one of *OSPF, BGP, RIB, BFD, NWV.*
uptime	How long the BFD session has been up; a string in the following format: *HH:MM:SS*.

Get BFD Sessions for the Specified Protocol

Get BFD sessions for the specified protocol.

Request

Method Type	GET
Request URI	/nos/api/info/bfd/application/<*protocol_name*>
Request Body (JSON)	

where:

Element	Description
protocol_name	(Mandatory) Protocol for which the BFD session is active (string); one of: *OSPF, BGP, RIB, BFD, NWV*.

Response

Response Body

(JSON)

```
[
   {
      "loc_addr": "<loc_addr>",
      "rem_addr": "<rem_addr>",
      "loc_disc": "<loc_disc>",
      "rem_disc": "<rem_disc>",
      "RH/RS":  "<RH/RS>",
      "holdown": "<holdown>",
      "mult": "<mult>",
      "sess_state":  "<sess_state>",
      "if_name":  "<if_name>"
    }
]
```

where:

Element	Description
`loc_addr`	BFD session source address (string); a valid IPv4 or IPv6 address.
`rem_addr`	BFD session destination IP address (string); a valid IPv4 or IPv6 address.
`loc_disc`	Unique number used by the local system to identify the BFD session; an integer from *1-2147483647*.
`rem_disc`	Unique number used by the remote system to identify the BFD session; an integer from *1-2147483647*.
`RH/RS`	Remote Heard/Remote State (string); one of *UP, DOWN, ADMIN_DOWN*.

Element	Description
`holdown`	If no BFD packet is received in the specified number of milliseconds, the session will be declared down; an integer from *150-2997*.
`mult`	The number of times a packet is missed before BFD declares the neighbor down; an integer from *3-50*.
`sess_state`	BFD session state (string); one of *UP*, *DOWN*, *ADMIN_DOWN*.
`if_name`	Interface on which the BFD session is active. **Note:** The interface must exist.

Get all BFD Session Details for the Specified Protocol

Get BFD session details for the specified protocol.

Request

Method Type	GET
Request URI	/nos/api/info/bfd/application_details/<*protocol_name*>
Request Body (JSON)	

where:

Element	Description
protocol_name	(Mandatory) Protocol for which the BFD session is active (string); one of: *OSPF, BGP, RIB, BFD, NWV.*

Response

Response Body

(JSON)

```
[
   {
      "loc_addr": <loc_addr>,
      "rem_addr": <rem_addr>,
      "loc_disc": <loc_disc>,
      "rem_disc": <rem_disc>,
      "RH/RS": <RH/RS>,
      "holdown": <holdown>,
      "mult": <mult>,
      "sess_state": <sess_state>,
      "if_name": <if_name>,
      "echo_mode": <echo_mode>,
      "gtsm": <gtsm>,
      "gtsm_ttl": <gtsm_ttl>,
      "minTxInt": <minTxInt>,
      "minRxInt": <minRxInt>,
      "negotiated minRxInt": <negotiated_minRxInt>,
      "negotiated multiplier": <negotiated_multiplier>,
      "rx Count": <rx_Count>,
      "tx Count": <tx_Count>,
      "registered protocols": <registered_protocols>,
      "uptime": <uptime>,
    }
]
```

where:

Element	Description
`loc_addr`	BFD session source address (string); a valid IPv4 or IPv6 address.
`rem_addr`	BFD session destination IP address (string); a valid IPv4 or IPv6 address.
`loc_disc`	Unique number used by the local system to identify the BFD session; an integer from *1-2147483647*.
`rem_disc`	Unique number used by the remote system to identify the BFD session; an integer from *1-2147483647*.
`RH/RS`	Remote Heard/Remote State (string); one of *UP*, *DOWN*, *ADMIN_DOWN*.
`holdown`	If no BFD packet is received in the specified number of milliseconds, the session will be declared down; an integer from *150-2997*.
`mult`	The number of times a packet is missed before BFD declares the neighbor down; an integer from *3-50*.
`sess_state`	BFD session state (string); one of *UP, DOWN, ADMIN_DOWN*.
`if_name`	Interface on which the BFD session is active. **Note:** The interface must exist.
`echo_mode`	Whether the BFD session has echo mode enabled (string); one of *enabled*, *disabled*.
`gtsm`	Whether the BFD session has Generalized TTL Security Mechanism (GTSM) enabled; one of *enabled*, *disabled*.
`gtsm_ttl`	Displays BFD GTSM Time-To Live (TTL) value; an integer from *1-255*.
`minTxInt`	Rate at which BFD control packets will be sent to BFD neighbors; an integer from *50-999*.
`minRxInt`	Specifies the rate at which BFD control packets will be expected to be received from BFD neighbors; an integer from *50-999*.
`negotiated minRxInt`	Specifies the negotiated rate at which BFD control packets will be received from BFD neighbors; an integer from *50-999*.
`negotiated multiplier`	The number of times a packet is missed before BFD declares the session down; an integer from *3-50*.
`rx Count`	The number of received BFD packets; a positive integer.
`tx Count`	The number of sent BFD packets; a positive integer.

Element	Description
`registered protocols`	Protocol for which the BFD session is active (string); one of *OSPF*, *BGP*, *RIB*, *BFD*
`uptime`	How long the BFD session has been up; a string in the following format: *HH:MM:SS*.

Get BFD Sessions with the Specified Local Address

Get BFD sessions with the specified local address.

Request

Method Type	GET
Request URI	/nos/api/info/bfd/loc_addr/<*loc_addr*>
Request Body (JSON)	

where:

Element	Description
loc_addr	(Mandatory) The BFD session source IP address (string); a valid IPv4 or IPv6 address.

Response

Response Body
 	
(JSON)
	[
	{
	"loc_addr": "<*loc_addr*>",
	"rem_addr": "<*rem_addr*>",
	"loc_disc": "<*loc_disc*>",
	"rem_disc": "<*rem_disc*>",
	"RH/RS": "<*RH/RS*>",
	"holdown": "<*holdown*>",
	"mult": "<*mult*>",
	"sess_state": "<*sess_state*>",
	"if_name": "<*if_name*>"
	}
	]

where:

Element	Description
`loc_addr`	BFD session source address (string); a valid IPv4 or IPv6 address.
`rem_addr`	BFD session destination IP address (string); a valid IPv4 or IPv6 address.
`loc_disc`	Unique number used by the local system to identify the BFD session; an integer from *1-2147483647*.
`rem_disc`	Unique number used by the remote system to identify the BFD session; an integer from *1-2147483647*.
`RH/RS`	Remote Heard/Remote State (string); one of *UP, DOWN, ADMIN_DOWN*.

Element	Description
`holdown`	If no BFD packet is received in the specified number of milliseconds, the session will be declared down; an integer from *150-2997*.
`mult`	The number of times a packet is missed before BFD declares the neighbor down; an integer from *3-50*.
`sess_state`	BFD session state (string); one of *UP, DOWN, ADMIN_DOWN*.
`if_name`	Interface on which the BFD session is active. **Note:** The interface must exist.

Get BFD Sessions with the Specified Remote Address

Get BFD sessions with the specified remote address.

Request

Method Type	GET
Request URI	/nos/api/info/bfd/rem_addr/<*rem_addr*>
Request Body (JSON)	

where:

Element	Description
rem_addr	(Mandatory) The BFD session destination IP address (string); a valid IPv4 or IPv6 address.

Response

Response Body

(JSON)

```
[
   {
      "loc_addr":  "<loc_addr>",
      "rem_addr":  "<rem_addr>",
      "loc_disc": "<loc_disc>",
      "rem_disc": "<rem_disc>",
      "RH/RS":  "<RH/RS>",
      "holdown": "<holdown>",
      "mult": "<mult>",
      "sess_state":  "<sess_state>",
      "if_name":  "<if_name>"
   }
]
```

where:

Element	Description
`loc_addr`	BFD session source address (string); a valid IPv4 or IPv6 address.
`rem_addr`	BFD session destination IP address (string); a valid IPv4 or IPv6 address.
`loc_disc`	Unique number used by the local system to identify the BFD session; an integer from *1-2147483647*.
`rem_disc`	Unique number used by the remote system to identify the BFD session; an integer from *1-2147483647*.
`RH/RS`	Remote Heard/Remote State (string); one of *UP, DOWN, ADMIN_DOWN*.

Element	Description
`holdown`	If no BFD packet is received in the specified number of milliseconds, the session will be declared down; an integer from *150-2997*.
`mult`	The number of times a packet is missed before BFD declares the neighbor down; an integer from *3-50*.
`sess_state`	BFD session state (string); one of *UP, DOWN, ADMIN_DOWN*.
`if_name`	Interface on which the BFD session is active. **Note:** The interface must exist.

Change BFD Global Options

Set or unset BFD global options.

Request

Method Type	POST
Request URI	/nos/api/cfg/bfd

Request Body
 	
(JSON)
	[
	{
	“min_tx”: "*<min_tx>*”,
	“min_rx” : "*<min_rx>*”,
	“multiplier”: "*<multiplier>*”,
	“ai_family” : "*<ai_family>*”,
	“slow_timer” : "*<slow_timer>*”,
	“gtsm_enable” : "*<gtsm_enable>*”,
	“gtsm_ttl” : "*<gtsm_ttl>*”
	}
	]

where:

Element	Description
`min_tx`	Desired rate at which BFD will be able to send packets to the BFD neighbors; the string default for the default value, or an integer from *50-999*.
`min_rx`	Desired rate at which BFD will be able to receive packets from the BFD neighbors; the string default for the default value, or an integer from *50-999*.
`multiplier`	Desired number of times a packet can be missed before BFD will declare the neighbor down; the string default for the default value, or an integer from *3-50*.
`ai_family`	The address family (string); one of *ipv4*, *ipv6*.
`slow_timer`	(Optional) Desired rate at which BFD will send control packets when the BFD session is down or when the BFD echo feature is enabled, in miliseconds; an integer from 1000-30000.
`gtsm_enable`	(Optional) Enable or disable GTSM protection; one of *true*, *false*.
`gtsm_ttl`	(Optional) Sets the desired BFD GTSM TTL; the string default for the default value, or an integer from *1-255*.

Note: You must submit values for the parameters `min_tx`, `min_rx`, and `multiplier` in the JSON body for the configuration to be applied. If you do not want to change the BFD options, omit these values from the JSON body.

Response

Response Body (JSON)	

Change BFD Interface Options

Set or unset BFD options for a specified interface.

Request

Method Type	POST
Request URI	/nos/api/cfg/bfd/interface/<*if_name*>

Request Body
 	
(JSON)
	[
	{
	"min_tx": "<*min_tx*>",
	"min_rx" : "<*min_rx*>",
	"multiplier": "<*multiplier*>",
	"ai_family" : "<*ai_family*>",
	"bfd_ipv4" : "<*bfd_ipv4*>",
	"bfd_ipv6" : "<*bfd_ipv6*>
	"echo_enable" : "<*echo_enable*>",
	"auth_type" : "<*auth_type*>",
	"auth_key_id" : "<*auth_key_id*>",
	"auth_key" : "<*auth_key*>",
	"auth_key_chain" : "<*auth_key_chain*>"
	}
	]

where:

Element	Description
if_name	(Mandatory) Interface name; a string. **Note:** The interface must exist.
`min_tx`	(Optional) Desired rate at which BFD will be able to send packets to the BFD neighbors; either `default` for the default value, or an integer from *50-999*.
`min_rx`	(Optional) Desired rate at which BFD will be able to receive packets from the BFD neighbors; either `default` for the default value, or an integer from *50-999*.
`multiplier`	(Optional) Desired number of times a packet can be missed before BFD will declare the neighbor down; the string default for the default value, or an integer from *3-50*.
`ai_family`	(Optional) The address family (string); one of *ipv4*, *ipv6*.
`bfd_ipv4`	(Optional) Enable or disable BFD for all IPv4 BFD sessions that use the named interface; one of `true`, `false`.
`bfd_ipv6`	(Optional) Enable or disable BFD for all IPv6 BFD sessions that use the named interface; one of *true*, *false*.
`echo_enable`	(Optional) Enable or disable BFD echo feature for all BFD sessions that use the named interface; one of *true*, *false*.

Element	Description
`auth_type`	(Optional) The authentication for the BFD session (string); one of *simple*, *keyed-md5*, *keyed-sha1*, *keyed-sha256*, *meticulous-keyed-md5*, *meticulous-keyed-sha1*, *meticulous-keyed-sha256*, *none*.
`auth_key_chain`	(Optional) Sets the authentication with key chain; a string (key chain name). **Note:** This value applies to all single-hop BFD sessions that use the named interface. If present in the JSON body, both `auth_key_id` and `auth_key` must be absent.
`auth_key_id`	(Optional) Sets the authentication key ID; an integer from *0-255*. **Note:** This value applies to all single-hop BFD sessions that use the named interface. If present in the JSON body, auth_key_id must also be present and auth_key_chain must be absent.
`auth_key`	(Optional) Sets the authentication key (string). **Note:** This value applies to all single-hop BFD sessions that use the named interface. If present in the JSON body, both `auth_key_id` and `auth_key` must be absent.

Note: You must submit values for the parameters `min_tx`, `min_rx`, and `multiplier` in the JSON body for the configuration to be applied. If you do not want to change the BFD options, omit these values from the JSON body.

Response

Response Body (JSON)	

Create a New BFD Session

Create a new BFD session.

Request

Method Type	POST
Request URI	/nos/api/cfg/bfd/interface/<*if_name*>/neighbors

Request Body
 	
(JSON)
	{
	“loc_addr” : "<*loc_addr*>”,
	“rem_addr” : "<*rem_addr*>”,
	“multihop” : "<*multihop*>”
	“admin_down” : "<*admin_down*>”
	“non_persistent” : "<*non_persistent*>”
	}

where:

Element	Description
if_name	(Mandatory) Interface on which the BFD session is active (string); a valid IPv4 or IPv6 address. **Note:** The interface must exist.
`local_addr`	(Mandatory) BFD session source IP address; a valid IPv4 or IPv6 address.
`rem_addr`	(Mandatory) BFD session destination IP address; a valid IPv4 or IPv6 address.
`multihop`	(Mandatory) Sets BFD session type: singlehop or multihop (string); one of *true*, *false*.
`admin_down`	(Optional) Set the BFD session state; *true* for down; *false* for up. Default value: *false*.
`non_persistent`	(Optional) Create the BFD session as non-persistent; *true* for non-persistent; *false* for persistent. Default value: *false*.

Response

Response Body (JSON)	

Delete a BFD Session

Delete an existing BFD session

Request

Method Type	DELETE
Request URI	/nos/api/cfg/bfd/interface/<*if_name*>/neighbors/<*loc_disc*>
Request Body (JSON)	

where:

Element	Description
if_name	(Mandatory) Interface on which the BFD session is active. **Note:** The interface must exist.
loc_disc	(Mandatory) Unique number used by the local system to identify the BFD session; an integer.

Response

`True` if the operation succeeded; otherwise `False`.

Set or Unset BFD Multihop Session Options

Set or unset BFD multihop session options

Request

Method Type	POST
Request URI	/nos/api/cfg/bfd/multihop_peer/<*rem_addr*>

Request Body
 	
(JSON)
	{
	"min_tx": "<*min_tx*>",
	"min_rx" : "<*min_rx*>",
	"multiplier": "<*multiplier*>",
	"auth_type" : "<*auth_type*>",
	"auth_key_id" : "<*auth_key_id*>",
	"auth_key" : "<*auth_key*>",
	"auth_key_chain" : "<*auth_key_chain*>"
	}

where:

Element	Description
rem_addr	(Mandatory) The IP address of the BFD neighbor; a valid IPv4 or IPv6 address.
`min_tx`	(Optional) Desired rate at which BFD will be able to send packets to the BFD neighbors; either `default` for the default value, or an integer from *50-999*.
`min_rx`	(Optional) Desired rate at which BFD will be able to receive packets from the BFD neighbors; either `default` for the default value, or an integer from *50-999*.
`multiplier`	(Optional) Desired number of times a packet can be missed before BFD will declare the neighbor down; either `default` for the default value, or an integer from *3-50*.
`auth_type`	(Optional) The authentication for the BFD session (string); one of *simple*, *keyed-md5*, *keyed-sha1*, *keyed-sha256*, *meticulous-keyed-md5*, *meticulous-keyed-sha1*, *meticulous-keyed-sha256*, *none*.
`auth_key_id`	(Optional) Sets the authentication key ID; an integer from *0-255*. **Note:** This value applies to all single-hop BFD sessions that use the named interface. If this value is present in the JSON body, `auth_key` must be present and `auth_key_chain` must be absent.
`auth_key`	(Optional) Sets the authentication key string; a string. **Note:** This value applies to all single-hop BFD sessions that use the named interface. If present in the JSON body, auth_key_id must also be present and auth_key_chain must be absent.

Element	Description
`auth_key_chain`	(Optional) Sets the authentication with key chain; a string (key chain name). **Note:** This value applies to all single-hop BFD sessions that use the named interface. If present in the JSON body, both `auth_key_id` and `auth_key` must be absent.

Response

Response Body (JSON)	

BGP

The following BGP URIs are available:

- /nos/api/cfg/bgp/global/stats GET, DELETE
- /nos/api/info/bgp/neighbor/adj_rib_in GET
- /nos/api/info/bgp/neighbor/adj_rib_out GET
- /nos/api/cfg/bgp/global GET
- /nos/api/cfg/bgp/bestpath GET
- /nos/api/cfg/bgp/confed GET
- /nos/api/cfg/bgp/graceful-restart GET
- /nos/api/cfg/bgp/route-reflector/<*vrf_name*>/ GET
- /nos/api/info/bgp/global/rib GET
- /nos/api/info/bgp/global/rib/details/l2vpn/ GET
- /nos/api/info/bgp/global/rib/network GET
- /nos/api/info/bgp/neighbor/summary GET
- /nos/api/info/bgp/neighbor/details/ GET
- /nos/api/info/bgp/neighbor/stats GET
- /nos/api/cfg/bgp/distance/<*af_name*>/<*saf_name*>/<*vrf_name*> GET
- /nos/api/cfg/bgp/af GET
- /nos/api/cfg/bgp/af/maximum_paths GET
- /nos/api/cfg/bgp/af/nht_delay GET
- /nos/api/cfg/bgp/af/aggregate GET
- /nos/api/cfg/bgp/af/dampening GET
- /nos/api/info/bgp/dampening/dampened_path GET
- /nos/api/cfg/bgp/af/network GET
- /nos/api/cfg/bgp/af/redistribute GET, PUT
- /nos/api/cfg/bgp/neighbor/details GET, PUT
- /nos/api/cfg/bgp/unnumbered PUT
- /nos/api/cfg/interface/bgp_unnumbered/<*if_name*> PUT

The following BGP interface property commands are available:

- Get BGP Global Statistics
- Clear BGP Global Statistics
- Get BGP Neighbor Received RIB Information
- Get BGP Neighbor RIB Advertised Information

- Get BGP Global Configuration
- Get BGP Best Path Configuration
- Get BGP Confederation Configuration
- Get BGP Graceful-Restart Configuration
- Get BGP Route Reflector Information
- Get BGP RIB Information
- Get BGP RIB Information for EVPN Routes
- Get BGP Detailed RIB Information
- Get BGP Summary Information
- Get BGP Neighbor Details
- Get BGP Neighbor Statistics
- Get BGP Distance Configuration
- Get BGP Address Family Global Configuration
- Get BGP Multipath ECMP Numbers Configuration
- Get BGP Nexthop Trigger-Delay Configuration
- Get BGP Aggregate Configuration
- Get BGP Dampening Parameters Configuration
- Get BGP Dampened Path Configuration
- Get BGP Network Configuration
- Get BGP Redistribute Configuration
- Set BGP Redistribute Configuration
- Get BGP Neighbor Configuration
- Set BGP Neighbor Configuration
- Update BGP Neighbor Configuration
- Update Global BGP Unnumbered Configuration
- Update BGP Unnumbered Interface Configuration

Get BGP Global Statistics

Gets global BGP statistics.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/stats/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	Virtual Routing and Forwarding name; one of the VRF name, "`default`," "`all`". Default value: `default`.

Response

Response Body

(JSON)

```
[
  {
    "vrf_name":"<vrf_name>",
    "stats": {
      "in_msgs": "<in_msgs>",
      "out_msgs ": "<sent_ msg>",
      "bytes_in": "<bytes_in>",
      "bytes_out": "<bytes_out>",
      "open_in ": "<open_in>",
      "open_out": "<open_out>",
      "update_in": "<update_in>",
      "update_out": "<update_out>",
      "keepalive_in": "<keepalive_in>",
      "keepalive_out": "<keepalive_out>",
      "notify_in": "<notify_in>",
      "notify_out": "<notify_out>",
      "refresh_in": "<refresh_in>",
      "refresh_out": "<refresh_out>",
      "dynamic_cap_in": "<dynamic_cap_in>",
      "dynamic_cap_out": "<dynamic_cap_out>",
    }
  }
]
```

where:

Element	Description
`vrf_name`	VRF name (string).
`in_msgs`	Received message number; a positive integer.
`out_msgs`	Send messge number; a positive integer.

Element	Description
bytes_in	Bytes received; a positive integer.
bytes_out	Bytes sent; a positive integer.
open_in	Open message input count; a positive integer.
open_out	Open message output count; a positive integer.
update_in	Update message input count; a positive integer.
update_out	Update message ouput count; a positive integer.
keepalive_in	Keepalive input count; a positive integer.
keepalive_out	Keepalive output count; a positive integer.
notify_in	Notify input count; a positive integer.
notify_out	Notify output count; a positive integer.
refresh_in	Route Refresh input count; a positive integer.
refresh_out	Route Refresh output count; a positive integer.
dynamic_cap_in	Dynamic Capability input count.; a positive integer.
dynamic_cap_out	Dynamic Capability output count; a positive integer.

Clear BGP Global Statistics

Clears global BGP statistics.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/bgp/stats/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	Virtual Routing and Forwarding name; one of the VRF name, "`default`," "`all`". Default value: `default`.

Response

Response Body (JSON)	

Get BGP Neighbor Received RIB Information

Gets the BGP neighbor received Routing Information Base information.

Request

Method Type	GET
Request URI	/nos/api/info/bgp/neighbor/adj_rib_in/<*neighbor_ip*>/<*af_name*>/<*vrf_name*>/<*subaf_name*>
Request Body (JSON)	

where:

Element	Description
neighbor_ip	Neighbor IP address; a valid IPv4 or IPv6 address.
vrf_name	(Optional) Address family name; one of `ipv4` or `ipv6m`. Default value: `ipv4`.
af_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.
subaf_name	(Optional) Subaddress family name; one of `unicast`, `multicast`. Default value: `unicast`.

Response

Response Body
 	
(JSON)
	[
	  {
	  "routes":
	  [
	  {
	  "origin": "*<origin>*",
	  "network": "*<network>*",
	  "mask_len": "*<mask_len>*",
	  "weight": "*<attr_weight>*",
	  "Metric": "*<metric>*",
	  "nexthop": "*<nexthop>*",
	  "aspath4B": "*<aspath4B>*",
	  "status": "*<falg>*",
	  "local_pref": "*<local_pref>*",
	  "aspath": "*<aspath>*",
	  }
	  ]
	  }
	]

where:

Element	Description
`origin`	Route origin attribute; one of: • `i` - IGP • `e` - EGP • `?` - incomplete
`network`	Route destination IP address; a valid IPv4 or IPv6 address.
`mask_len`	Route mask length; an integer from 0-32.
`weight`	Route weight attribute; an integer from 0-65535.
`metric`	Route Multi-Exit Discriminator attribute; an integer from 0~4294967295.
`nexthop`	Route next hop; a valid IP address.
`aspath4B`	Route 4B AS path; an AS path VTY string.
`status`	Router status; one of: • `s` - suppressed • `d` - damped • `h` - history • `*` - valid • `>` - best • `i` - internal
`local_pref`	Route local preference attribute; an integer from 0-4294967295.
`aspath`	Route AS path attribute; an AS path VTY string.

Get BGP Neighbor RIB Advertised Information

Gets information about the advertised BGP neighbor Routing Information Base.

Request

Method Type	GET
Request URI	/nos/api/info/bgp/neighbor/adj_rib_out/*<neighbor>*/*<af_name>*/ /*<vrf_name>*/*<subaf_name>*
Request Body (JSON)	

where:

Element	Description
neighbor	Neighbor IP address; a valid IPv4 or IPv6 address.
af_name	(Optional) Address family name; one of `ipv4` or `ipv6m`. Default value: `ipv4`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.
subaf_name	(Optional) Subaddress family name; one of `unicast`, `multicast`. Default value: `unicast`.

Response

Response Body
 	
(JSON)
	[
	  {
	  "routes":
	  [
	  {
	  "origin": "*<origin>*",
	  "network": "*<prefix_addr>*",
	  "mask_len": "*<mask_len>*",
	  "weight": "*<weight>*",
	  "Metric": "*<metric>*",
	  "nexthop": "*<nexthop>*",
	  "aspath4B": "*<aspath4B>*",
	  "status ": "*<status>*",
	  "local_pref": "*<local_pref>*",
	  "aspath": "*<aspath>*",
	  }
	  ]
	  }
	]

where:

Element	Description
`origin`	Route origin attribute; one of: • `i` - IGP • `e` - EGP • `?` - incomplete
`network`	Route destination IP address; a valid IPv4 or IPv6 address.
`mask_len`	Route mask length; an integer from 0-32.
`weight`	Route weight attribute; an integer from 0-65535.
`metric`	Route med attribute; an integer from 0-4294967295.
`nexthop`	Route nexthop address; a valid IPv4 or IPv6 address.
`aspath4B`	Route 4B AS path; an AS path VTY string.
`status`	Router status; one of: • `s` - suppressed • `d` - damped • `h` - history • `*` - valid • `>` - best • `i` - internal
`local_pref`	Route local preference attribute; an integer from 0-4294967295.
`aspath`	Route AS path attribute; a valid AS path VTY string.

Get BGP Global Configuration

Gets the BGP global configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/global/<*vrf_name*>/
Request Body (JSON)	

where:

Element	Description
vrf_name	VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	<pre>[
	  {
	  "vrf_name": "<vrf_name>",
	  "status": "<status>",
	  "router_id": "<router_id>",
	  "as_number": "<as_number>",
	  "keep-alive timer": "<keep-alive timer>",
	  "hold-down timer": "<hold-down timer>",
	  "as-local-count ": "<as-local-count>",
	  "enforce-first-as ": "<enforce-first-as>",
	  "fast-external-failover ": "<fast-external-failover>",
	  "log-neighbor-changes ": "<log-neighbor-changes>",
	  "maxas-limit ": "<maxas-limit>",
	  "synchronization ": "<synchronization>"
	  }
	]</pre>

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, `default`, `all`. Default value: `default`.
`status`	BGP global status; one of disable, enable.
`router_id`	BGP router ID; a valid IPv4 or IPv6 address.
`as_number`	BGP AS number; an integer from 1-4294967295.
`keep-alive timer`	Keep alive interval, in seconds; an integer from 0-3600.
`hold-down timer`	Hold time, in seconds; an integer from 0-3600.

Element	Description
`as-local-count`	Number of times the local AS is to be prepended; an integer from 1-64.
`enforce-first-as`	Enforce the first AS for EBGP routes; one of `enable`, `disable`.
`fast-external-failover`	Immediately reset session if a link to a directly connected external peer goes down; one of `enable`, `disable`.
`log-neighbor-changes`	Log reasons for neighbor going up, down, and resetting; one of `enable`, `disable`.
`maxas-limit`	Allow the AS-PATH attribute from EBGP neighbor to impose a limit on the number of ASes; an integer from 0-2000.
`synchronization`	Perform IGP synchronization; one of `enable`, `disable`.

Get BGP Best Path Configuration

Gets the BGP best path configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/bestpath/<*vrf_name*>/
Request Body (JSON)	

where:

Element	Description
vrf_name	VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	` {`
	`  "vrf_name": "`*<vrf_name>*`",`
	`  "always-compare-med ": "`*<always-compare-med>*`",`
	`  "as-path-ignore":"`*<as-path-ignore>*`",`
	`  "as-path-multipath-relax": "`*<as-path-multipath-relax>*`",`
	`  "compare-confed-aspath": "`*<compare-confed-aspath>*`",`
	`  "compare-routerid ": "`*<compare-routerid>*`",`
	`  "dont-compare-originator-id": "`*<dont-compare-originator-id>*`",`
	`  "med-confed ": "<med-confed>",`
	`  "med-missing-as-worst": "`*<med-missing-as-worst>*`",`
	`  "med-non-deterministic": "`*<med-non-deterministic>*`",`
	`  "med-remove-recv-med ": "`*<med-remove-recv-med>*`",`
	`  "med-remove-send-med ": "`*<med-remove-send-med>*`",`
	`  "tie-break-on-age": "`*<tie-break-on-age>*`"`
	` }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default, all*. Default value: *default*.
`always-compare-med`	Allow comparing MED from different neighbors; one of *enable, disable*.
`as-path-ignore`	Ignore as-path length in selecting a route; one of *enable, disable*.
`as-path multipath-relax`	Relax AS-Path restriction when choosing multipaths; one of *enable, disable*.

Element	Description
`compare-confed-as path`	Allow comparing confederation AS path length; one of *enable, disable*.
`compare-routerid`	Compare router IDs for identical EBGP paths; one of *enable, disable*.
`dont-compare-originator-id`	Don't compare originator IDs for BGP; one of *enable, disable*.
`med-confed`	Compare MED among confederation paths; one of *enable, disable*.
`med-missing-as-worst`	Treat missing MED as the least preferred one; one of *enable, disable*.
`med-non-deterministic`	Best MED path among paths not selected from same AS; one of *enable, disable*.
`med-remove-recv-med`	Whether to remove received MED attribute; one of *enable, disable*.
`med-remove-send-med`	Whether to remove send MED attribute; one of *enable, disable*.
`tie-break-on-age`	Whether to prefer the old route when `compare-route-id` is not set; one of *enable, disable*.

Get BGP Confederation Configuration

Gets the BGP confederation configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/confed/<*vrf_name*>/
Request Body (JSON)	

where:

Element	Description
vrf_name	VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	` {`
	`  "vrf_name": "`<*vrf_name*>`",`
	`  "identifier ": "`<*identifier*>`",`
	`  "peers": "`<*peers*>`",`
	` }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default, all*. Default value: *default*.
`identifier`	Routing domain confederation AS; an integer from 0-65535.
`peers`	Peer ASes in BGP confederation; an integer from 1-65535.

Get BGP Graceful-Restart Configuration

Gets the BGP graceful-restart configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/graceful-restart/<*vrf_name*>/
Request Body (JSON)	

where:

Element	Description
vrf_name	VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	` {`
	`  "vrf_name": "`<*vrf_name*>`",`
	`  "stalepath-time ": "`<*stalepath-time*>`",`
	`  "helper-status": "`<*helper-status*>`",`
	` }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default, all*. Default value: *default*.
`stalepath-time`	The delay value, in seconds, to remove routes marked as stale; an integer from 1-3600.
`helper-status`	Status of Graceful Restart Helper Mode functionality; one of *enabled, disabled*.

Get BGP Route Reflector Information

Gets BGP route reflector information.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/rr/*<vrf_name>*/
Request Body (JSON)	

where:

Element	Description
vrf_name	VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	` {`
	`  "vrf_name": "`*<vrf_name>*`",`
	`  "cluster-id ": "`*<cluster-id>*`",`
	` }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default, all*. Default value: *default*.
`cluster-id`	Route reflector cluster ID; a valid IP address.

Get BGP RIB Information

Gets BGP Routing Information Base Information.

Request

Method Type	GET
Request URI	/nos/api/info/bgp/global/rib/*<af_name>*/*<vrf_name>*/
Request Body (JSON)	

where:

Element	Description
af_name	(Optional) Address family name; one of `ipv4`, `ipv6`,`l2vpn`. Default value; both.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body

(JSON)

```
[
   {
      "status": "{status code}",
      "network": "{network}",
      "nextHopGlobal": "{global next-hop}",
      "nextHopLocal": "{local next-hop}",
      "Metric": "{metric}",
      "local_pref": "{local preference}",
      "weight": "{weight}",
      "pathInfo": "{path infofmation}",
      "medvalue": "{med value}",
      "med": "{med}",
      "aspath": "{as path string}",
      "aspath4B": "{4B as path string}",
      "origin": "{route origin}"
   },
```

where:

Element	Description
`status`	Router status code; one of: • `s` - suppressed • `d` - damped • `h` - history • `*` - valid • `>` - best • `i` – internal

Element	Description
`network`	Route destination IP address; a valid IPv4 or IPv6 address. For l2vpn: • `EVPN type-1 prefix: [1]:[ESI]:[EthTag]` • `EVPN type-2 prefix: [2]:[ESI]:[EthTag]:[MAClen]:[MAC]`
`nextHopGlobal`	Route nexthop IPv6 address. Not used for IPv4.
`nextHopLocal`	Route nexthop; a valid IPv4 or IPv6 address.
`weight`	Route weight attribute; an integer from 0-65535.
`pathInfo`	Route path information; a valid AS path VTY string.
`medvalue`	Multi-exit discriminator value if the MED attribute is missing and missing-as-worst is set; an integer from 0-4294967294.
`med`	Multi-exit discriminator value; an integer from 0-4294967294.
`aspath`	Route AS path attribute; a valid AS path VTY string.
`aspath4B`	Route 4B AS path; a valid AS path VTY string.
`origin`	Route origin attribute; one of the following: • `i` - IGP • `e` - EGP • `?` - incomplete

Get BGP RIB Information for EVPN Routes

Gets BGP Routing Information Base Information.

Request

Method Type	GET
Request URI	nos/api/info/bgp/global/rib/details/l2vpn/<*keyword*>/<*value*>/<*vni*>/<*rd*>
Request Body (JSON)	

where:

Element	Description
keyword	(Mandatory) A string describing the route type; one of `esi`, `mac`.
value	(Mandatory) A MAC or Ethernet segment ID in the following format: `EEEE.EEEE.EEEE/ESI`.
vni	(Mandatory) The virtual network identifier; an integer from 1 -16777216.
rd	(Optional) The route distinguisher, in the following format: `IP_address:nn`.

Response

Response Body

(JSON)

```
[
   {
      "table entry for":"{prefix}"
      "paths":"{count}"
      "best":"{prefix}"
      "no advertise":"{prefix}"
      " no export":"{prefix}"
      " local as":"{prefix}"
      " suppress":"{prefix}"
      "adv non peer-group":"{ non_peer_group}"
      "adv peer-group":"{peer_group}"
      "no peer adv":"{peer_no_val}"
      {
        "as path str":"{as_path_str}"
        "aggregator as":"{aggregator_as}"
        "aggregator as4":"{aggregator_as4}"
        "aggregator address":"{aggregator_addr}"
        "Rec from RR-client":"{af_flag}"
        "suppressed (damp)":"{rt_state_is_damp}"
        "history entry":"{ rt_state_is_history}"
        " nexthop address":"{nexthop}"
        "peer":"{peer}"
        "inaccessible":"{flag}"
        "igpmetric":"{igpmetric}"
        "from peer":"{peer_su_str}"
        "orig id":"{org_id}"
        "next-hop local ip":"{nexthop_local_ipv6}"
        "metric removed":"{metric_removed}"
        "local pref":"{local_pref}"
        "weight":"{weight}"
        "label":"{label}"
        "valid":"{valid}"
        "stale":"{stale}"
        "multipath-candidate":"{flag_misc}"
        "installed":"{flag_misc}"
        "synchronized":"{flag_misc}"
        "atomic aggregate":"{flags_atomic_agg}"
        "best":"{flag}"
        "commnunity":"{community}"
        "extended community":"{ecommunity}"
        "originator":"{originator_id}"
        "cluster id":"{cluster_id}"
        "reuse info":"{reuse_info}"
        "last update":"{curr_time}"
      }
   },
]
```

where:

Element	Description
`table entry for`	Route IP address/mask; a valid IP address and net mask.
`paths`	The number of paths to destination.
`best`	Whether this is the best path; one of *Yes, No*.

Element	Description
`no advertise`	Not advertised to any peers; one of *Yes, No*.
`no export`	Not advertised to EBGP peers; one of *Yes, No*.
`local as`	The local AS number, from 1-4294967295.
`suppress`	Whether advertisements are suppressed by an aggregate; one of *Yes, No*.
`adv non peer-group`	The non-peer group name.
`adv peer-group`	The peer group name.
`no peer adv`	Not advertised to any peer; one of *Yes, No*.
`as path str`	Route AS path attribute; a valid AS path VTY string.
`aggregator as`	Aggregator AS number.
`aggregator as4`	Aggregator 4-byte AS number.
`aggregator address`	Aggregator address.
`rec from RR-client`	Received from RR-client; one of *Yes, No*.
`suppressed (damp)`	Suppressed due to dampening; one of *Yes, No*.
`history entry`	History entry; one of *Yes, No*.
`nexthop address`	Route nexthop; a valid IP address.
`peer`	Peer address
`inaccessible`	Whether the RIB is can be accessed; one of *Yes, No*.
`igpmetric`	IGP metric value.
`from peer`	The peer address.
`orig id`	The originator ID.
`next-hop local ip`	The nexthop IP address, a valid IP address.
`metric removed`	Whether the metric is removed; one of *Yes, No*.
`local pref`	Local preference value.
`weight`	Route weight attribute; an integer from 0-65535.
`label`	The label.

Element	Description
`valid`	Whether the path is valid; one of *Yes, No*.
`stale`	Whether the state is stale; one of *Yes, No*.
`multipath-candidate`	Whether this is a multipath candidate; one of *Yes, No*.
`installed`	Whether installed; one of *Yes, No*.
`synchronized`	Whether synchronized; one of *Yes, No*.
`atomic aggregate`	Whether this is as atomic aggregate; one of *Yes, No*.
`best`	Whether this is the best path; one of *Yes, No*.
`community`	The community string.
`extended community`	The extended community string.
`originator`	Originator ID.
`cluster id`	Cluster ID.
`reuse info`	Reuse information.
`last update`	Last update time.

Get BGP Detailed RIB Information

Gets detailed BGP Routing Information Base information.

Request

Method Type	GET
Request URI	/nos/api/info/bgp/global/rib/network/<*af_name*>/<*route*>/<*network_mask*>/<*vrf_name*>/
Request Body (JSON)	

where:

Element	Description
af_name	(Optional) Address family name; one or both of `ipv4`, `ipv6`. Default value: both.
route	Route; a valid IPv4 or IPv6 address.
network_mask	Network mask: • IPv4: An integer from 0-32. • IPv6: An integer from 0-128.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	[
	  {
	  "table entry for": "*<table entry for>*"
	  "paths": [
	  {
	  "as path str": "*<as path str>*"
	  "aggregator as": "*<aggregator as>*"
	  "aggregator as4": "*<aggregator as4>*"
	  "aggregator address": "*<aggregator address>*"
	  "Rec from RR-client": "*<Rec from RR-client>*"
	  "suppressed (damp)": "*<suppressed (damp)>*"
	  "history entry": "*<history entry>*"
	  "nexthop address": "*<nexthop address>*"
	  "peer": "*<peer>*"
	  "inaccessible": "*<inaccessible>*"
	  "igpmetric": "*<igpmetric>*"
	  "from peer": "*<from peer>*"
	  "orig id": "*<orig id>*"
	  "next-hop local ip": "*<next-hop local ip>*"
	
	  "origin": "*<origin>*"
	  "metric": "*<metric>*"
	  "local pref": "*<local pref>*"
	  "weight": "*<weight>*"
	  "label": "*<label>*"
	  "valid": "*<valid>*"
	  "stale": "*<stale>*"
	  "type": "*<type>*"
	  "multipath-candidate": "*<multipath-candidate>*"
	  "installed": "*<installed>*"
	  "synchronized": "*<synchronized>*"
	  "atomic aggregate": "*<atomic aggregate>*"
	  "best": "*<best>*"
	  "commnunity": "*<community>*"
	  "extended community": "*<extended community>*"
	  "originator": "*<originator>*"
	  "cluster-id": "*<cluster-id>*"
	  "reuse info": "*<reuse info>*"
	  "last update": "*<last update>*"
	  }
	  ],
	  "best is no.": "*<best is no.>*"
	  "advertised to any peer": "*<advertised to any peer>*"
	  "advertised to EBGP peer": "*<advertised to EBGP peer>*"
	  "advertised outside local AS ": "*<Yes/No>*"
	  "advertisements suppressed by an aggregate": "*<advertisements suppressed by an aggregate>*"
	  "advertised to non peer-group peers": "*<advertised to non peer-group peers>*"
	  "advertised to peer-groups": "*<advertised to peer-groups>*"
	  "not advertised": "*<not advertised>*"
	  },
	]

where:

Element	Description
`table entry for`	Route IP address/mask; a valid IP address and net mask.
`paths`	Dictionary; marks the beginning of path table for specific route entry.
`as path str`	Route path information; a valid AS path VTY string.
`aggregator as`	Aggregator AS number.
`aggregator as4`	Aggregator 4-byte AS number.
`aggregator address`	Aggregator address.
`Rec from RR-client`	Received from RR-client; *Yes*. **Note:** This value only appears if it has been set.
`suppressed (damp)`	Suppressed due to dampening; *Yes*. **Note:** This value only appears if it has been set.
`history entry`	History entry; `Yes`. **Note:** This value only appears if it has been set.
`nexthop address`	Route nexthop; a valid IPv4 or IPv6 address.
`peer`	Peer address.
`inaccessible`	Whether the RIB is can be accessed; one of *Yes*, *No*.
`igpmetric`	IGP metric value; No. **Note:** This value only appears if it is *No*.
`from peer`	Whether the from peer address can be accessed, *No*. **Note:** This value only appears if it is *No*.
`orig id`	Whether the originator ID can be accessed; *No*.
`next-hop local ip`	Whether the next-hop IP address can be accessed; a valid IP address or *No*. **Note:** The value `No` only appears if it is inaccessible.
`metric`	Metric; one of the metric value, `removed`.
`local pref`	Local preference value; only appears if set.
`weight`	Route weight attribute; an integer from 0-65535. **Note:** This value only appears if it is set.
`label`	Label; only appears if set.

Element	Description
`valid`	Whether the path is valid; *Yes*. **Note:** This value only appears if the path is valid.
`stale`	Whether the state is stale; *Yes*. **Note:** This value only appears if the state is stale.
`multipath-candidate`	Whether this is a multipath candidate; one of *Yes*, *No*.
`installed`	Whether installed; one of *Yes*, *No*.
`synchronized`	Whether synchronized; one of *Yes*, *No*.
`atomic aggregate`	Whether this is an atomic aggregate; *Yes*. **Note:** This value only appears if it is *Yes*.
`best`	Whether this is the best path; one of *Yes*, *No*.
`community`	Community string.
`extended community`	Extended community string.
`originator`	Originator ID.
`cluster-id`	Cluster ID.
`reuse info`	Reuse information.
`last update`	Last update time.
`best is no.`	Which path number is best; the maximum number of paths for this destination.
`advertised to any peer`	Whether this is advertised to any peers; one of *Yes*, *No*.
`advertised to EBGP peer`	Whether this is advertised to an EBGP peer; one of *Yes*, *No*.
`advertised outside local AS`	Whether this is advertised outside the local AS; one of *Yes, No*.
`advertisements suppressed by an agregate`	Whether advertisements are suppressed by an aggregate; one of *Yes, No*.
`advertised to non peer-group peers`	IP address advertised to non peer-group peers.
`advertised to peer-groups`	IP address advertised to peer groups.

Element	Description
`not advertised`	Not advertised to any peer. **Note:** This value only appears if *true*.

Get BGP Summary Information

Gets BGP summary information.

Request

Method Type	GET
Request URI	/nos/api/info/bgp/neighbor/summary/<*vrf_name*>/<*af_name*> /<*subaf_name*>/
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.
af_name	(Optional) Address family name; one of `ipv4`, `ipv6`,`l2vpn`. Default value: `ipv4`.
subaf_name	Subsequent Address Family Identifier name; `unicast`,`epvn`. Default value: `unicast`.

Response

Response Body
 	
(JSON)
	<pre>[
	  {
	  "router id":"{router_id}"
	  "table version":"{table_version}"
	  "path count":"{aspath_count}"
	  "conf max ebgp paths":"{cfg_maxpath_ebgp}"
	  "max ebgp paths":"{maxpath_ebgp}"
	  "conf max ibgp paths":"{cfg_maxpath_ibgp}"
	  "max ibgp paths":"{maxpath_ibgp}"
	  "local AS count":"{local_as_count}"
	  "peer":"{host_name}"
	  "peer version":"{version}"
	  "peer AS":"{peer_as}"
	  "open in":"{open_in}"
	  "update in":"{update_in}"
	  "keepalive in":"{keepalive_in}"
	  "refresh in":"{refresh_in}"
	  "dynamic cap in":"{dynamic_cap_in}"
	  "open out":"{open_out}"
	  "update out":"{update_out}"
	  "keepalive out":"{keepalive_out}"
	  "refresh out":"{refresh_out}"
	  "dynamic cap out":"{dynamic_cap_out}"
	  },
	]</pre>

where:

Element	Description
`router id`	Router ID; a valid IPv4 or IPv6 address.
`table version`	The table version.
`path count`	The path number.
`conf max ebgp paths`	The maximum configured EBGP paths.
`max ebgp paths`	The maximum EBGP paths.
`conf max ibgp paths`	The maximum configured IBGP paths.
`max ibgp paths`	The maximum IBGP paths.
`local AS count`	The local AS count.
`peer`	Peer address; a valid IPv4 or IPv6 address.
`peer version`	Peer version.
`peer AS`	Peer AS.
`open in`	Number of received open messages.
`update in`	Number of received updates.
`keepalive in`	Number of received keepalives.

Element	Description
`refresh in`	Number of received route refresh.
`dynamic cap in`	Dynamic capabilities input count.
`open out`	Number of sent open messages.
`update out`	Number of sent updates.
`keepalive out`	Number of sent keepalive messages.
`refresh out`	Number of sent route refresh messages.
`dynamic cap out`	Dynamic capabilities output count.

Get BGP Neighbor Details

Gets BGP neighbor detailed information.

Request

Method Type	GET
Request URI	/nos/api/info/bgp/neighbor/details/<*nbr-ip*>/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
nbr-ip	(Optional) Neighbor IP address; a valid IPv4 or IPv6 address.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body

(JSON)

```
[
   {
      "neighbor": "<neighbor>"
      "vrfname": "<vrfname>"
      "remote AS": "<remote AS>"
      "local AS": "<local AS>"
      "address family": "<address family>"
      "table version": "<table version>"
      "neighbor version": "<neighbor version>"
      "index val": "<index val>"
      "index offset": "<index offset>"
      "index mask": "<index mask>"
      "link type": "<link type>"
      "version": "<version>"
      "description": "<description>"
      "remote router-ID": "<remote router-ID>"
      "admin": "<admin>"
      "ifbound": "<ifbound>"
      "state": "<state>"
      "dyncap_adv": "<dyncap_adv>"
      "dyncap_rec": "<dyncap_rec>"
      "refresh_adv": "<refresh_adv>"
      "refresh_new_rec": "<refresh_new_rec>"
      "refresh_old_rec": "<refresh_old_rec>"
      "ext_asn_adv": "<ext_asn_adv>"
      "ext_asn_rec": "<ext_asn_rec>"
      "afc_adv": "<afc_adv>"
      "afc_recv": "<afc_recv>"
      "afc_VPN_adv": "<afc_VPN_adv>"
      "afc_VPN_recv": "<afc_VPN_recv>"
      "afc_mcast_adv": "<afc_mcast_adv>"
      "afc_mcast_recv": "<afc_mcast_recv>"
```

Response Body
 	
(JSON-continued)
	<pre>"uptime": "<uptime>"
	"peer-group name": "<peer-group name>"
	"holdtime": "<holdtime>"
	"keepalive": "<keepalive>"
	"conf holdtime": "<conf holdtime>"
	"conf keepalive": "<conf keepalive>"
	"recvMsg": "<recvMsg>"
	"recvNotf": "<recvNotf>"
	"recvQueue": "<recvQueue>"
	"sentMsg": "<sentMsg>"
	"sentNotf": "<sentNotf>"
	"sentQueue": "<sentQueue>"
	"refresh_in": "<refresh_in>"
	"refresh_out": "<refresh_out>"
	"routeadv": "<routeadv>"
	"update_if": "<update_if>"
	"update_source": "<update_source>"
	"established": "<established>"
	"dropped": "<dropped>"
	"prefix overflow": "<prefix overflow>"
	"ttl": "<ttl>"
	"local address": "<local address>"
	"local port": "<local port>"
	"remote address": "<remote address>"
	"remote port": "<remote port>"
	"nextHopAddress": "<nextHopAddress>"
	"nextHopLocalV6": "<nextHopLocalV6>"
	"nextHopGlobalV6": "<nextHopGlobalV6>"
	"shared_network": "<shared_network>"
	"next conn retry": "<next conn retry>"
	"err notif": "<err notif>"
	"last_reset_time": "<last_reset_time>"
	"error code": "<error code>"
	"error subcode": "<error subcode>"
	"rmap_map": "<rmap_map>"
	},
	]</pre>

where:

Element	Description
neighbor	Neighbor address.
vrfname	VRF name.
remote AS	AS number.
local AS	Local AS number.
address family	Address family.
table version	Table version.
neighbor version	Neighbor index value.
index val	Index offset.
index offset	Index mask.
index mask	Link type; one of *internal, external*.

Element	Description
link type	Link type; one of *internal*, *external*.
version	Version.
description	Description.
remote router-ID	Remote router ID.
admin	Admin state.
ifbound	Whether the interface is bound; one of *No interface binding*, *Interface bound*.
state	Neighbor state.
dyncap_adv	Dynamic capability advertised, only if advertised.
dyncap_rec	Dynamic capability received, only if received.
refresh_adv	Refresh capability advertised, only if advertised.
refresh_new_rec	Refresh New received, only if received.
refresh_old_rec	Refresh Old received, only if received.
ext_asn_adv	Extended ASN capability advertised.
ext_asn_rec	Extended ASN capability received.
afc_adv	Address family unicast sent.
afc_recv	Address family unicast received.
afc_VPN_adv	Address family VPN sent.
afc_VPN_recv	Address family VPN received.
afc_mcast_adv	Address family multicast sent.
afc_mcast_recv	Address family multicast received.
uptime	Uptime.
peer-group name	Peer IP address.
holdtime	Holdtime.
keepalive	Keepalive time.
conf holdtime	Configured holdtime.
conf keepalive	Configured keepalive time.
recvMsg	Number of received messages.
recvNotf	Number of received notifications.
recvQueue	Received messages queue count.
sentMsg	Number of sent messages.

Element	Description
sentNotf	Number of sent notifications.
sentQueue	Sent messages queue count.
refresh_in	Number of route refresh messages received.
refresh_out	Number of route refresh messages sent.
routeadv	Number of router advertisements.
update_if	Update interface.
update_source	Update source address.
established	Established count.
dropped	Dropped count.
prefix overflow	Whether there is a prefix overflow; one of *Yes, No.*
ttl	Time to live value.
local address	Local neighbor IP address.
local port	Local port number.
remote address	Remote peer IP address.
remote port	Remote port number.
nextHopAddress	Next-hop address.
nextHopLocalV6	Next-hop address (link local).
nextHopGlobalV6	Next-hop address (global).
shared network	Shared network.
next conn retry	Number of retries.
err notif	Whether there was an error notification; one of *sent, received.*
last_reset_time	Last reset time.
err code	Code string.
err subcode	Subcode string.
rmap_name	Default originating route map.

Get BGP Neighbor Statistics

Gets BGP neighbor detailed statistics.

Request

Method Type	GET
Request URI	/nos/api/info/bgp/neighbor/stats/<*nbr-ip*>/<*item*>/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
nbr-ip	IP address; one or more valid IPv4 or IPv6 addresses. Default; all neighbor IP addresses.
item	The type of statistics; one or more of `keepalive`, `notification`, `open`, `update`, `recv_msgs`, `sent_msgs`. Default; show all items.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "statistic type": "`*<statistic>*`"`
	`  "received": "`*<received>*`"`
	`  "sent": "`*<sent>*`"`
	`  }`
	`]`

where:

Element	Description
`statistic type`	Statistic type; one or more of *keepalive*, *notification*, *open*, *update*, *recv_msgs*, *sent_msgs*.
`received`	Number of received messages of the specified type or types.
`sent`	Number of sent messages of the specified type or types.

Get BGP Distance Configuration

Gets the BGP distance configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/distance/<*af_name*>/<*saf_name*> /<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "vrf_name": "`*<vrf_name>*`"`
	`  "distance_ebgp": "`*<distance_ebgp>*`"`
	`  "distance_ibgp": "`*<distance_ibgp>*`"`
	`  "distance_local": "`*<distance_local>*`"`
	`  }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default*, *all*. Default value: *default*.
`distance_ebgp`	Distance for routes external to the AS; an integer from 0-255.
`distance_ibgp`	Distance for routes internal to the AS; an integer from 0-255.
`distance_local`	Distance for routes local to the AS; an integer from 0-255.

Get BGP Address Family Global Configuration

Gets the BGP address family global configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/af/<*af_name*>/<*saf_name*>/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "vrf_name": "`<*vrf_name*>`"`
	`  "cc_reflection": "`<*cc_reflection*>`"`
	`  "synchronization": "`<*synchronization*>`"`
	`  "network_synchronization": "`<*network_synchronization*>`"`
	`  }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default*, *all*. Default value: *default*.
`cc_reflection`	Client-to-client reflect; one of *enable*, *disable*.
`synchronization`	Perform IGP synchronization; one of *enable*, *disable*.
`network_ synchronization`	Perform IGP synchronization on network routes; one of *enable*, *disable*.

Get BGP Multipath ECMP Numbers Configuration

Gets the BGP multipath maximum ECMP numbers configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/af/maximum_paths/*<af_name>*/*<saf_name>* /*<vrf_name>*
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "vrf_name": "`*<vrf_name>*`"`
	`  "ibgp_max_number": "`*<ibgp_max_number>*`"`
	`  "ebgp_max_number": "`*<ebgp_max_number>*`"`
	`  }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default*, *all*. Default value: *default*.
`ibgp_max_number`	IBGP multipath maximum ECMP number; an integer from 1-32.
`ebgp_max_number`	EBGP multipath maximum ECMP number; an integer from 1-32.

Get BGP Nexthop Trigger-Delay Configuration

Gets the BGP nexthop trigger-delay configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/af/nht_delay/<*af_name*>/<*saf_name*>
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "critical": "`<*critical*>`"`
	`  "non-critical": "`<*noncritical*>`"`
	`  }`
	`]`

where:

Element	Description
`critical`	Nexthop changes affecting reachability; an integer from 1-4294967295.
`non-critical`	Nexthop changes affecting metric; an integer from 1-4294967295.

Get BGP Aggregate Configuration

Gets the BGP aggregate configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/af/aggregate/*<af_name>*/*<saf_name>*/*<vrf_name>*
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "vrf_name": "`*<vrf_name>*`"`
	`  "prefix": "`*<prefix>*`"`
	`  "type": "`*<type>*`"`
	`  }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default*, *all*. Default value: *default*.
`prefix`	Aggregate prefix; an IP address in one of the following forms: • A.B.C.D/M • X:X::X:X/M.
`type`	Aggregate type; one of the following: • *as_set* - Generate AS set path information. • *summary_only* - Filter more specific routes from updates. • *as_set_summary_only* - Both *as-set* and *summary-only.*

Get BGP Dampening Parameters Configuration

Gets the BGP dampening parameters configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/af/dampening/*<af_name>*/*<saf_name>* /*<vrf_name>*
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "vrf_name ": "`*<vrf_name>*`",`
	`  "half_life ": "`*<half_life>*`",`
	`  "reuse_penalty ": "`*<reuse_penalty>*`",`
	`  "suppress_penalty ": "`*<suppress_penalty>*`",`
	`  "max_suppress ": "`*<max_suppress>*`",`
	`  "unreach_half_life ": "`*<unreach_half_life>*`",`
	`  "rmap_name ": "`*<rmap_name>*`"`
	`  }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default*, *all*. Default value: *default*.
`half_life`	Reachability half-life time for the penalty, in minutes; an integer from 1-45.
`reuse_penalty`	Value to start reusing a route; an integer from 1-20000.
`suppress_ penalty`	Value to start suppressing a route; an integer from 1-20000.

Element	Description
`max_suppress`	Maximum duration to suppress a stable route, in minutes; an integer from 1-255.
`unreach_half_life`	Unreachability half-life time for the penalty, in minutes; an integer from 1-45.
`rmap_name`	Route-map name; a string up to 63 characters long.

Get BGP Dampened Path Configuration

Gets the BGP dampened path configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/af/dampening/dampened_path/*<af_name>*/*<saf_name>*/*<vrf_name>*
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body

(JSON)

```
[
  {
   "statusCode": "<statusCodeVal>",
   "network": "<networkVal>",
   "nextHopGlobal": "<nextHopGlobalVal>",
   "nextHopLocal": "<nextHopLocalVal>",
   "metric": "<metricVal>",
   "pathInfo": "<pathInfoVal>",
   "reuseTime": "<reuseTimeVal>",
   "asPathStr": "<asPathStrVal>",
   "asPath4BStr": "<asPath4BStrVal>",
   "routeOriginType": "<routeOriginTypeVal>"
  }
]
```

where:

Element	Description
`statusCode`	Router status code; one of: • `s` - suppressed • `d` - damped • `h` - history • `*` - valid • `>` - best • `i` – internal
`network`	Route destination IP address; a valid IPv4 or IPv6 address.
`nextHopGlobal`	Route nexthop IPv6 address. Not used for IPv4.
`nextHopLocal`	Route nexthop; a valid IPv4 or IPv6 address.
`metric`	Route metric; an integer.
`pathInfo`	Route path information; a valid AS path VTY string.
`reuseTime`	Route reuse time; a string.
`asPathStr`	Route AS path attribute; a valid AS path VTY string.
`asPath4BStr`	Route 4B AS path; a valid AS path VTY string.
`routeOrigin Type`	Route origin type; one of the following: • `i` - IGP • `e` - EGP • `?` - incomplete

Get BGP Network Configuration

Gets the BGP network configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/af/network/<*af_name*>/<*saf_name*>/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "vrf_name ": "`<*vrf_name*>`",`
	`  "prefix ":  "`<*prefix*>`",`
	`  "backdoor":  "`<*backdoor*>`",`
	`  "rmap_name":  "`<*rmap_name*>`"`
	`  }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default*, *all*. Default value: *default*.
`prefix`	Network prefix; an IP address in one of the following forms: • A.B.C.D/M • X:X::X:X/M.
`backdoor`	Whether a BGP backdoor route is specified; one of *enable*, *disable*.
`rmap_name`	Route map name; a string up to 63 characters long.

Get BGP Redistribute Configuration

Gets the BGP network redistribute configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/af/redistribute/<*af_name*>/<*saf_name*> /<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
af_name	Address family name; one or both of `ipv4`, `ipv6`. Default value; both.
saf_name	Subsequent Address Family Identifier name; `unicast`. Default value: `unicast`.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	[
	{
	"vrf_name ": "<*vrf_name*>",
	"redist_direct": "<*redist_direct*>",
	"direct_rmap_name ": "<*direct_rmap_name*>",
	"redist_ospf": "<*redist_ospf*>",
	"ospf_rmap_name ": "<*ospf_rmap_name*>",
	"redist_static ": "<*redist_static*>",
	"static_rmap_name ": "<*static_rmap_name*>"
	}
	]

where:

Element	Description
`vrf_name`	VRF name; one of the VRF name, *default*, *all*. Default value: *default*.
`redist_direct`	Whether redistribute direct is enabled; one of *enable*, *disable*.
`direct_rmap_name`	Route map name for redistribute direct; a string up to 63 characters long.

Element	Description
`redist_ospf`	Whether redistribute OSPF is enabled; one of *enable*, *disable*.
`ospf_rmap_name`	Route map name for redistribute OSPF; a string up to 63 characters long.
`redist_static`	Whether redistribute static is enabled; one of *enable*, *disable*.
`static_rmap_name`	Route map name for redistribute static; a string up to 63 characters long.

Set BGP Redistribute Configuration

Sets the BGP address family redistribute configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/bgp/af/redistribute/<*afi*>/<*safi*>

Request Body
 	
(JSON)
	[
	{
	“redist_host_info”: ”enable”
	}
	]

where:

Element	Description
af_name	Address family name; one of `ipv4`, `ipv6`, `l2vpn`. Default value: all.
subaf_name	Subsequent Address Family Identifier name; `unicast`, `evpn`. Default value: `unicast`.

Response

Response Body
 	
(JSON)
	[
	{
	“redist_host_info”: ”enable”
	}
	]

where:

Element	Description
`redist_host_info`	Redistribution for host information; one of *enable*, *disable*.

Get BGP Neighbor Configuration

Gets the BGP network neighbor configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/bgp/neighbor/details/<*neighbor_ip*>/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
neighbor_ip	(Optional) The IP address of the neighbor; a valid IPv4 or IPv6 address. No value will display all neighbors.
vrf_name	(Optional) VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	<pre>[
	  {
	    "neighbor": "<*neighbor*>",
	    "vrfname": "<*vrfname*>",
	    "remote as": "<*remote as*>",
	    "local as": "<*local as*>",
	    "address family": "<*address family*>",
	    "advertisement interval": "<*advertisement interval*>",
	    "bfd": "<*bfd*>",
	    "connection retry time": "<*connection retry time*>",
	    "description": "<*description*>",
	    "disallow infinite holdtime": "<*disallow infinite holdtime*>",
	    "do not capability negociate": "<*do not capability negociate*>",
	    "advertise dynamic capability": "<*advertise dynamic capability*>",
	    "EBGP multihop": "<*EBGP multihop*>",
	    "remote private as": "<*remote private as*>",
	    "maximum peers": "<*maximum peers*>",
	    "password": "<*password*>",
	    "shutdown": "<*shutdown*>",
	    "peer holdtime": "<*peer holdtime*>",
	    "peer keepalive": "<*peer keepalive*>",
	    "connection-mode passive": "<*connection-mode passive*>",
	    "ttl security hops": "<*ttl security hops*>",
	    "update-source": "<*update-source*>",
	    "weight": "<*weight*>",
	    "allow as in": "<*allow as in*>",
	    "default originate": "<*default originate*>",
	    "default originate rmap": "<*default originate rmap*>",
	    "prefix-list in": "<*prefix-list in*>",
	    "prefix-list out": "<*prefix-list out*>",
	    "maximum-prefix": "<*maximum-prefix*>",
	    "maximum-prefix warning": "<*maximum-prefix warning*>",
	   "maximum-prefix threshold percent":"<*maximum-prefix threshold percent*>",
	    "next-hop-self": "<*next-hop-self*>",
	    "filter-list in": "<*filter-list in*>",
	    "filter-list out": "<*filter-list out*>",
	    "route-map in": "<*route-map in*>",
	    "route-map out": "<*route-map out*>",
	    "route reflector client": "<*route reflector client*>",
	    "send community": "<*send community*>",
	    "send community extended": "<*send community extended*>",
	    "soft reconfiguration inbound": "<*soft reconfiguration inbound*>",
	    "unsuppress-map": "<*unsuppress-map*>"
	  }
	]</pre>

where:

Element	Description
`neighbor`	Neighbor IP address; a valid IPv4 or IPv6 address.
`vrf_name`	VRF name; one of the VRF name, *default*, *all*. Default value: *default*.
`remote as`	Current neighbor AS; an AS number.
`local as`	Switch AS; an AS number.

Element	Description
`address family`	Neighbor address family; an address family.
`advertisement interval`	Minimum interval between BGP updates, in seconds; an integer.
`bfd`	BFD state; one of: • *enabled* • *disabled* • *multihop enabled*
`connection retry time`	BGP connect timer, in seconds; an integer.
`description`	Neighbor description; a string.
`disallow infinite holdtime`	Neighbor disallow infinite hold time; one of *Yes*, *No*.
`do not capability negociate`	Whether to perform capability negotiations; one of *Yes*, *No*.
`advertise dynamic capability`	Advertise dynamic capability to this neighbor; one of *Yes*, *No*.
`EBGP multihop`	Number of multihops; an integer from 1 - 255.
`remote private as`	Whether to remove private AS number from outbound packets; one of *Yes*, *No*.
`maximum peers`	Maximum number of peers for this prefix; an integer from 1-96.
`password`	Neighbor password; an encrypted password.
`shutdown`	Neighbor state; one of *Yes*, *No*.
`peer holdtime`	Holdtime value, in seconds; an integer from 0-3600.
`peer keepalive`	Keepalive value, in seconds; an integer from 0-3600.
`connection-mode passive`	Whether to allow a passive connection; one of *Yes*, *No*.
`ttl security hops`	Number of hops; an integer from 1-254.
`update-source`	Source of routing updates; one of: • *ethernet* • *vlan* • *loopback interfaces*
`weight`	The default weight for routes from this neighbor; an integer from 0–65535.

Element	Description
`allow as in`	Accept AS path with own AS in it; an integer from 1-10.
`default originate`	Whether to originate default route to this neighbor; one of *Yes*, *No*.
`default originate rmap`	Route map that specifies criteria; an RMAP name.
`prefix-list in`	Filter updates from this neighbor; a filter name.
`prefix-list out`	Filter updates to this neighbor; a filter name.
`maximum-prefix`	Maximum number of prefix accept from this peer; an integer from -15782.
`maximum-prefix warning`	Whether to only give a warning message when limit is exceeded; one of *Yes*, *No*.
`maximum-prefix threshold percent`	Threshold value; an integer from 1-100.
`next-hop-self`	Whether to disable the next hop calculation for this neighbor; one of *Yes*, *No*.
`filter-list in`	Establish filter for incoming routes; a filter name.
`filter-list out`	Establish filter for outgoing routes; a filter name.
`route-map in`	Apply routemap for incoming routes; an RMAP name.
`route-map out`	Apply routemap for outgoing routes; an RMAP name.
`route reflector client`	Whether to set neighbor as route reflector client; one of *Yes*, *No*.
`send community`	Whether to send community attribute to this neighbor; one of *Yes*, *No*.
`send community extended`	Whether to send extended community attribute to this neighbor; one of *Yes*, *No*.
`soft reconfiguration inbound`	Whether to allow inbound soft reconfiguration for this neighbor; one of *Yes*, *No*.
`unsuppress-map`	Route map to selectively unsuppress suppressed routes; an RMAP name.

Set BGP Neighbor Configuration

Sets the BGP network neighbor configuration details. If a neighbor does not exist, it will be created.

Request

Method Type	PUT
Request URI	/*nos*/*api*/*cfg*/*bgp*/*neighbor*/*details*/<*neighbor_ip*>/<*mask*>/vrf/<*vrf_name*>

Request Body
 	
(JSON)
	`[`
	`  {`
	`  "address family":{afi safi value}`
	`  }`
	`]`

where:

Element	Description
neighbor_ip	(Mandatory) The IP address of the neighbor; a valid IPv4 or IPv6 address.
mask	The group mask: • 1-32 for IPv4 • 1-128 for IPv6
vrf	The VFR instance name; must be specified before <*vrf_name*>.
vrf_name	(Optional) VRF name; one of the VRF name, `default`. Default value: `default`.
address family	The Address Family Identifier or Subsequent Address Family Identifier value; `ipv4 unicast`, `ipv6 unicast`, `l2vpn evpn`. **Note:** When this is configured, send extended community is automatically set.

Response

Response Body
 	
(JSON)
	<pre>[
	  {
	" neighbor ":"{ip_addr }",
	" vrfname": "{ vrf }",
	" remote as ": "{ as value of neighbor }",
	" local as": "{ local as value }",
	" address family ": "{ address family type }",
	" advertisement interval ": "{ advertisement interval value
	}",
	" bfd ": "{ enabled/disabled/multihop enabled }",
	"connection retry time" : "{ connection retry time vaue }",
	"description" : "{ peer description }",
	"disallow infinite holdtime" : "{ yes/no }",
	"do not capability negociate" : "{ yes/no }",
	"advertise dynamic capability" : "{ yes/no }",
	"EBGP multihop" : "{ multihop value }",
	"remote private as" : "{ yes/no }",
	"maximum peers" : "{ maximum peers value }",
	"password" : "{ encrypted password }",
	"shutdown" : "{ yes/no }",
	"peer holdtime" : "{ holdtime value }",
	"peer keepalive" : "{ keepalive value }",
	"connection-mode passive" : "{ yes/no }",
	"ttl security hops" : "{ number of hops }",
	"update-source" : "{ eupdate source }",
	"weight" : "{ weight value }",
	"allow as in" : "{ number of as allowed }",
	"default originate" : "{ yes/no }",
	"default originate rmap" : "{ rmap name }",
	"prefix-list in" : "{ prefix list name }",
	"prefix-list out" : "{ prefix list name }",
	"maximum-prefix" : "{ max prefix value }",
	"maximum-prefix warning" : "{ yes/no }",
	"maximum-prefix threshold percent" : "{ percent value }",
	"next-hop-self" : "{ yes/no }",
	"filter-list in" : "{ list name }",
	"filter-list out" : "{ list name }",
	"route-map in" : "{ route map name }",
	"route-map out" : "{ route map name }",
	"route reflector client" : "{ yes/no }",
	"send community" : "{ yes/no }",
	"send community extended" : "{ yes/no }",
	"soft reconfiguration inbound" : "{ yes/no }",
	"unsuppress-map" : "{ route map name }"
	  }
	]</pre>

where:

Element	Description
`neighbor`	Neighbor IP address; a valid IPv4 or IPv6 address.
`vrf_name`	VRF name; one of the VRF name, *default*. Default value: *default*.
`remote as`	Current neighbor AS; an AS number.
`local as`	Switch AS; an AS number.

Element	Description
`address family`	Neighbor address family; an address family.
`advertisement interval`	Minimum interval between BGP updates, in seconds; an integer.
`bfd`	BFD state; one of: • *enabled* • *disabled* • *multihop enabled*
`connection retry time`	BGP connect timer, in seconds; an integer.
`description`	Neighbor description; a string.
`disallow infinite holdtime`	Neighbor disallow infinite hold time; one of *Yes*, *No*.
`do not capability negociate`	Whether to perform capability negotiations; one of *Yes*, *No*.
`advertise dynamic capability`	Advertise dynamic capability to this neighbor; one of *Yes*, *No*.
`EBGP multihop`	Number of multihops; an integer from 1 - 255.
`remote private as`	Whether to remove private AS number from outbound packets; one of *Yes*, *No*.
`maximum peers`	Maximum number of peers for this prefix; an integer from 1-96.
`password`	Neighbor password; an encrypted password.
`shutdown`	Neighbor state; one of *Yes*, *No*.
`peer holdtime`	Holdtime value, in seconds; an integer from 0-3600.
`peer keepalive`	Keepalive value, in seconds; an integer from 0-3600.
`connection-mode passive`	Whether to allow a passive connection; one of *Yes*, *No*.
`ttl security hops`	Number of hops; an integer from 1-254.
`update-source`	Source of routing updates; one of: • *ethernet* • *vlan* • *loopback interfaces*
`weight`	The default weight for routes from this neighbor; an integer from 0–65535.
`allow as in`	Accept AS path with own AS in it; an integer from 1-10.

Element	Description
`default originate`	Whether to originate default route to this neighbor; one of *Yes*, *No*.
`default originate rmap`	Route map that specifies criteria; an RMAP name.
`prefix-list in`	Filter updates from this neighbor; a filter name.
`prefix-list out`	Filter updates to this neighbor; a filter name.
`maximum-prefix`	Maximum number of prefix accept from this peer; an integer from -15782.
`maximum-prefix warning`	Whether to only give a warning message when limit is exceeded; one of *Yes*, *No*.
`maximum-prefix threshold percent`	Threshold value; an integer from 1-100.
`next-hop-self`	Whether to disable the next hop calculation for this neighbor; one of *Yes*, *No*.
`filter-list in`	Establish filter for incoming routes; a filter name.
`filter-list out`	Establish filter for outgoing routes; a filter name.
`route-map in`	Apply routemap for incoming routes; an RMAP name.
`route-map out`	Apply routemap for outgoing routes; an RMAP name.
`route reflector client`	Whether to set neighbor as route reflector client; one of *Yes*, *No*.
`send community`	Whether to send community attribute to this neighbor; one of *Yes*, *No*.
`send community extended`	Whether to send extended community attribute to this neighbor; one of *Yes*, *No*.
`soft reconfiguration inbound`	Whether to allow inbound soft reconfiguration for this neighbor; one of *Yes*, *No*.
`unsuppress-map`	Route map to selectively unsuppress suppressed routes; an RMAP name.

Update BGP Neighbor Configuration

Updates the configuration of a specific BGP neighbor. If the BGP neighbor is inexistent, it will be created.

Request

Method Type	PUT
Request URI	/nos/api/cfg/bgp/neighbor/details/<*IP*>/<*mask*>/vrf/<*vrf_name*>

Request Body (JSON)	[
	{
	"remote as":"{*as value of neighbor*}",
	"local as":"{*local as value*}",
	"advertisement interval":"{*advertisement interval value*}",
	"bfd":"{enabled/disabled/multihop enabled}",
	"connection retry time":"{*connection retry time value*}",
	"description":"{*peer description*}",
	"disallow infinite holdtime":"{yes/no}",
	"do not capability negociate":"{yes/no}",
	"advertise dynamic capability":"{yes/no}",
	"EBGP multihop":"{*multihop value*}",
	"remote private as":"{yes/no}",
	"maximum peers":"{*maximum peers value*}",
	"password":"{*encrypted password*}",
	"unencrypt-password":"{yes/no}",
	"shutdown":"{yes/no}",
	"peer holdtime":"{*holdtime value*}",
	"peer keepalive":"{*keepalive value*}",
	"connection-mode passive":"{yes/no}",
	"ttl security hops":"{*number of hops*}",
	"update-source":"{*update source*}",
	"weight":"{*weight value*}",
	"allow as in":"{*number of as allowed*}",
	"default originate":"{yes/no}",
	"default originate rmap":"{*rmap name*}",
	"prefix-list in":"{*prefix list name*}",
	"prefix-list out":"{*prefix list name*}",
	"maximum-prefix":"{*max prefix value*}",
	"maximum-prefix warning":"{yes/no}",
	"maximum-prefix threshold percent":"{*percent value*}",
	"next-hop-self":"{yes/no}",
	"filter-list in":"{*list name*}",
	"filter-list out":"{*list name*}",
	"route-map in":"{*route map name*}",
	"route-map out":"{*route map name*}",
	"route reflector client":"{yes/no}",
	"send community":"{yes/no}",
	"send community extended":"{yes/no}",
	"soft reconfiguration inbound":"{yes/no}",
	"unsuppress-map":"{*route map name*}"
	}
	]

where:

Element	Description
neighbor_ip	The IP address of the BGP neighbor.
vrf_name	The VRF instance for the BGP neighbor.
`remote as`	The current neighbor AS number; an integer from 1-4294967295.
`local as`	The current local AS number; an integer from 1-4294967295.
`advertisement interval`	The minimum time interval, in seconds, between consecutive BGP updates; an integer from 1-65535.
`bfd`	The status of Bidirectional Forwarding Detection (BFD). Valid values: • *enabled* • *disable* • *multihop enabled*
`connection retry time`	The connection retry time, in seconds; an integer from 1-65535.
`description`	The BGP neighbor description (string).
`disallow infinite holdtime`	Whether the configuration of infinite hold-time is disallowed; one of *yes*, *no*.
`do not capability negociate`	Whether capability negotiations are disabled; one of *yes*, *no*.
`advertise dynamic capability`	Whether dynamic capability advertisements are enabled; one of *yes*, *no*.
`EBGP multihop`	The number of EBGP multi-hops, an integer from 1-255.
`remote private as`	Whether the removal of private AS numbers from outbound routes updates is enabled; one of *yes*, *no*.
`maximum peers`	The maximum number of peers configured for the prefix of the BGP neighbor, an integer from 1-96.
`password`	The encrypted password for the BGP neighbor (sting).
`unencrypt-password`	Whether the password for the BGP neighbor is unencrypted; one of *yes*, *no*.
`shutdown`	Shuts down the BGP neighbor; one of *yes*, *no*.
`peer holdtime`	The time interval, in seconds, the switch awaits before transitioning the BGP neighbor to IDLE state, if the switch doesn't receive an update or keepalive message from the neighbor; an integer from 0-3600.

Element	Description
`peer keepalive`	The time interval, in seconds, the switch awaits before sending another keep-alive message to the BGP neighbor; an integer from 0-3600.
`connection-mode passive`	Whether the initiations of TCP sessions with the BGP neighbor are disabled; one of *yes*, *no*.
`ttl security hops`	The minimum number of TTL router hops an IP packet must have to not be discarded; one of *yes*, *no*.
`update-source`	The source of the BGP session and updates; one of: • *ethernet port* • *VLAN interface* • *loopback interface*
`weight`	The default weight of routes incoming from the BGP neighbor; an integer from 1-65535.
`allow as in`	The number of AS paths with the local AS number that are accepted by the switch; an integer from 1-10.
`default originate`	Whether to distribute a default route to the BGP neighbor; one of *yes*, *no*.
`default originate rmap`	The name of the route map for the default route.
`prefix-list in`	The prefix list for routes incoming from the BGP neighbor.
`prefix-list out`	The prefix list for routes outgoing to the BGP neighbor.
`maximum-prefix`	The maximum number of prefixes that can be received from the BGP neighbor, an integer from 1-15872.
`maximum-prefix warning`	Whether warning messages are generated only when the maximum prefix limit is exceeded; one of *yes*, *no*.
`maximum-prefix threshold percent`	The percentage of the maximum prefix limit at which the switch starts to generate a warning message; an integer from 1-100.
`next-hop-self`	Whether next-hop calculations for the BGP neighbor are disabled; one of *yes*, *no*.
`filter-list in`	The AS path ACL for routes incoming from the BGP neighbor. Default value: the ACL name.
`filter-list out`	The AS path ACL for routes outgoing to the BGP neighbor. Default value: the ACL name.
`route-map in`	The route map for routes incoming from the BGP neighbor. Default value: the route map name.

Element	Description
`route-map out`	The route map for routes outgoing to the BGP neighbor. Default value: the route map name.
`route reflector client`	Whether the BGP neighbor is a route reflector client; one of *yes*, *no*.
`send community`	Whether community attributes are sent to the BGP neighbor; one of *yes*, *no*.
`send community extended`	Whether extended community attributes are sent to the BGP neighbor; one of *yes*, *no*.
`soft reconfiguration inbound`	Whether the switch stores BGP neighbor updates; one of *yes*, *no*.
`unsuppress-map`	The route map to selectively unsuppress suppressed routes. Default value: the route map name.

Response

Response Body (JSON)	[
	{
	"neighbor":"{*ip_addr*}",
	"vrf_name":"{*vrf_name*}",
	"remote as":"{*as value of neighbor*}",
	"local as":"{*local as value*}",
	"address family":"{*address family type*}",
	"advertisement interval":"{*advertisement interval value*}",
	"bfd":"{enabled/disabled/multihop enabled}",
	"connection retry time":"{*connection retry time value*}",
	"description":"{*peer description*}",
	"disallow infinite holdtime":"{yes/no}",
	"do not capability negociate":"{yes/no}",
	"advertise dynamic capability":"{yes/no}",
	"EBGP multihop":"{*multihop value*}",
	"remote private as":"{yes/no}",
	"maximum peers":"{*maximum peers value*}",
	"password":"{*encrypted password*}",
	"shutdown":"{yes/no}",
	"peer holdtime":"{*holdtime value*}",
	"peer keepalive":"{*keepalive value*}",
	"connection-mode passive":"{yes/no}",
	"ttl security hops":"{*number of hops*}",
	"update-source":"{*update source*}",
	"weight":"{*weight value*}",
	"allow as in":"{*number of as allowed*}",
	"default originate":"{yes/no}",
	"default originate rmap":"{*rmap name*}",
	"prefix-list in":"{*prefix list name*}",
	"prefix-list out":"{*prefix list name*}",
	"maximum-prefix":"{*max prefix value*}",
	"maximum-prefix warning":"{yes/no}",
	"maximum-prefix threshold percent":"{*percent value*}",
	"next-hop-self":"{yes/no}",
	"filter-list in":"{*list name*}",
	"filter-list out":"{*list name*}",
	"route-map in":"{*route map name*}",
	"route-map out":"{*route map name*}",
	"route reflector client":"{yes/no}",
	"send community":"{yes/no}",
	"send community extended":"{yes/no}",
	"soft reconfiguration inbound":"{yes/no}",
	"unsuppress-map":"{*route map name*}"
	}
	]

where:

Element	Description
`neighbor_ip`	The IP address of the BGP neighbor.
`vrf_name`	The VRF instance for the BGP neighbor. Default value: the VRF instance name.
`remote as`	The current neighbor AS number; an integer from 1-4294967295.
`local as`	The current local AS number; an integer from 1-4294967295.
`address family`	The BGP neighbor address family; one of: • *IPv4* • *IPv6*
`advertisement interval`	The minimum time interval, in seconds, between consecutive BGP updates; an integer from 1-65535.
`bfd`	The status of Bidirectional Forwarding Detection (BFD); one of: • *enabled* • *disable* • *multihop enabled*
`connection retry time`	The connection retry time, in seconds; an integer from 1-65535.
`description`	The BGP neighbor description (string).
`disallow infinite holdtime`	Whether the configuration of infinite hold-time is disallowed; one of *yes*, *no*.
`do not capability negociate`	Whether capability negotiations are disabled; one of *yes*, *no*.
`advertise dynamic capability`	Whether dynamic capability advertisements are enabled; one of *yes*, *no*.
`EBGP multihop`	The number of EBGP multi-hops; an intefer from 1-255.
`remote private as`	Whether the removal of private AS numbers from outbound routes updates is enabled; one of *yes*, *no*.
`maximum peers`	The maximum number of peers configured for the prefix of the BGP neighbor; an integer from 1-96.
`password`	The encrypted password for the BGP neighbor (string).
`shutdown`	Shuts down the BGP neighbor; one of *yes*, *no*.

Element	Description
`peer holdtime`	The time interval, in seconds, the switch awaits before transitioning the BGP neighbor to IDLE state, if the switch doesn't receive an update or keepalive message from the neighbor; an integer from 0-3600.
`peer keepalive`	The time interval, in seconds, the switch awaits before sending another keep-alive message to the BGP neighbor; an integer from 0-3600.
`connection-mode passive`	Whether the initiations of TCP sessions with the BGP neighbor are disabled; one of *yes*, *no*.
`ttl security hops`	The minimum number of TTL router hops an IP packet must have to not be discarded; an integer from 1-254.
`update-source`	The source of the BGP session and updates; one of: • *ethernet port* • *VLAN interface* • *loopback interface*
`weight`	The default weight of routes incoming from the BGP neighbor; an integer from 1-65535.
`allow as in`	The number of AS paths with the local AS number that are accepted by the switch; an integer from 1-10.
`default originate`	Whether to distribute a default route to the BGP neighbor; one of *yes*, *no*.
`default originate rmap`	The name of the route map for the default route. Valid value: the route map name.
`prefix-list in`	The prefix list for routes incoming from the BGP neighbor. Valid value: the prefix list name.
`prefix-list out`	The prefix list for routes outgoing to the BGP neighbor. Valid value: the prefix list name.
`maximum-prefix`	The maximum number of prefixes that can be received from the BGP neighbor; an integer from 1-15872.
`maximum-prefix warning`	Whether warning messages are generated only when the maximum prefix limit is exceeded; one of *yes*, *no*.
`maximum-prefix threshold percent`	The percentage of the maximum prefix limit at which the switch starts to generate a warning message; an integer from 1-100.
`next-hop-self`	Whether next-hop calculations for the BGP neighbor are disabled; one of *yes*, *no*.
`filter-list in`	The AS path ACL for routes incoming from the BGP neighbor. Valid value: the ACL name.
`filter-list out`	The AS path ACL for routes outgoing to the BGP neighbor. Valid value: the ACL name.

Element	Description
`route-map in`	The route map for routes incoming from the BGP neighbor. Valid value: the route map name.
`route-map out`	The route map for routes outgoing to the BGP neighbor. Valid value: the route map name.
`route reflector client`	Whether the BGP neighbor is a route reflector client; one of *yes*, *no*.
`send community`	Whether community attributes are sent to the BGP neighbor; one of *yes*, *no*.
`send community extended`	Whether extended community attributes are sent to the BGP neighbor; one of *yes*, *no*.
`soft reconfiguration inbound`	Whether the switch stores BGP neighbor updates; one of *yes*, *no*.
`unsuppress-map`	The route map to selectively unsuppress suppressed routes. Valid value: the route map name.

Update Global BGP Unnumbered Configuration

Updates the global BGP unnumbered configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/bgp/unnumbered

Request Body (JSON)	`{`
	`"state":"{enabled|disabled}"`
	`"bfd":"{enabled|disabled}"`
	`}`

where:

Element	Description
`state`	The status of the global BGP unnumbered feature; one of: *enabled*, *disabled*.
`bfd`	(Optional) The BFD feature status for all unnumbered neighbors; one of: *enabled*, *disabled*. Default value: *disabled*.

Response

`True` if the operation succeeded; otherwise `False`.

Update BGP Unnumbered Interface Configuration

Updates the BGP unnumbered configuration for a specific switch interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/interface/bgp_unnumbered/<*if_name*>

Request Body (JSON)	{
	"state":"{enabled|disabled}"
	}

where:

Element	Description
if_name	The name of the switch interface, for example, *Ethernet1/12*.
`state`	The status of the global BGP unnumbered feature; one of: *enabled*, *disabled*.

Response

`True` if the operation succeeded; otherwise `False`.

CEE

The following Converged Enhanced Ethernet (CEE) URIs are available:

- /nos/api/cfg/cee GET, PUT
- /nos/api/cfg/cee/pfc GET, PUT
- /nos/api/cfg/cee/pfc/<*if_name*> GET, PUT
- /nos/api/info/cee/pfc/counters/<*if_name*> GET
- /nos/api/cfg/cee/ets GET, PUT
- /nos/api/cfg/cee/app-proto GET, POST, DELETE
- /nos/api/cfg/cee/dcbx/<*if_name*> GET, PUT
- /nos/api/info/cee/dcbx/ctrl/<*if_name*> GET
- /nos/api/info/cee/dcbx/admin/<*if_name*> GET
- /nos/api/info/cee/dcbx/oper/<*if_name*> GET
- /nos/api/info/cee/dcbx/remote/<*if_name*> GET

The following CEE commands are available:

- Get CEE Configuration
- Update CEE Configuration
- Get PFC Configuration
- Update PFC Configuration
- Get PFC Interface Configuration
- Update PFC Interface Configuration
- Get PFC Interface Statistics
- Get ETS Configuration
- Update ETS Configuration
- Get Application Protocol Configuration
- Create Application Protocol Configuration
- Delete Application Protocol Configuration
- Get DCBX Configuration
- Update DCBX Configuration
- Get DCBX Control Interface Information
- Get DCBX Administrative Interface Information
- Get DCBX Operational Interface Information
- Get DCBX Remote Interface Information

Get CEE Configuration

Gets the current CEE configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/cee
Request Body (JSON)	

Response

Response Body (JSON)	`{`
	`  "status":"{off|on}",`
	`}`

where:

Element	Description
`status`	The status of the CEE service on the switch; one of *off (disabled)*, *on (enabled)*. Default value: *off*.

Update CEE Configuration

Updates the CEE configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/cee

Request Body (JSON)	`{`
	`  "status":"{off|on}",`
	`}`

where:

Element	Description
`status`	The status of the CEE service on the switch; one of *off (disabled)*, *on (enabled)*. Default value: *off*.

Response

Response Body (JSON)	`{`
	`  "status":"{off|on}",`
	`}`

where:

Element	Description
`status`	The status of the CEE service on the switch; one of *off (disabled)*, *on (enabled)*. Default value: *off*.

Get PFC Configuration

Gets the current Priority Flow Control (PFC) configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/cee/pfc
Request Body (JSON)	

Response

Response Body (JSON)	{
	"state":"{off|on}",
	“priority_map”:”{*priority_map*}”
	}

where:

Element	Description
`state`	The status of PFC on the switch; one of *off (disabled), on (enabled)*. Default value: *on*.
`priority_map`	The PFC priority flow map. Valid value: the list of enabled priorities: 0-7. Default value: *3*.

Update PFC Configuration

Updates the PFC configuration.

Note: Unlike the CLI command, the PUT operation overwrites any previous configurations.

Request

Method Type	PUT
Request URI	/nos/api/cfg/cee/pfc

Request Body (JSON)	{
	"state":"{off|on}",
	“priority_map”:”{*priority_map*}”
	}

where:

Element	Description
`state`	The status of PFC on the switch; one of *off (disabled)*, *on (enabled)*. Default value: *on*.
`priority_map`	The PFC priority flow map. Valid value: the list of enabled priorities: 0-7. Default value: *3*. **Notes:** • Up to two PFC priorities can be simultaneously enabled, except for the NE10032, where the limit is one priority. • The PFC priority flow map must not contain a combination of enabled and disabled priorities. Configure ETS appropriately and then enabled PFC to avoid configuration issues.

Response

Response Body (JSON)	{
	"state":"{off|on}",
	“priority_map”:”{*priority_map*}”
	}

where:

Element	Description
`state`	The status of PFC on the switch; one of *off (disabled)*, *on (enabled)*.
`priority_map`	The PFC priority flow map. Valid value: the list of enabled priorities: 0-7.

Get PFC Interface Configuration

Gets the current PFC configuration for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/cee/pfc/{*if_name*}
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. Valid values: • *Interface name* (for example, *Ethernet1/12*)

Response

Response Body (JSON)	{
	"if_name":"{*if_name*}",
	"state":"{off|on}",
	}

where:

Element	Description
`if_name`	The name of the switch interface, for example: *Ethernet1/12*.
`state`	The status of PFC on the switch; one of: *off (disabled)*, *on (enabled)*.

Update PFC Interface Configuration

Updates the PFC configuration for a specific switch interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/cee/pfc/{*if_name*}

Request Body (JSON)	{
	"if_name":"{*if_name*}",
	"state":"{off|on}",
	}

where:

Element	Description
if_name	The name of the switch interface, for example: *Ethernet1/12*.
`state`	The status of PFC on the switch; one of: *off (disabled)*, *on (enabled)*. Default value: *on*.

Response

Response Body (JSON)	{
	"if_name":"{*if_name*}",
	"state":"{off|on}",
	}

where:

Element	Description
`if_name`	The name of the switch interface, for example: *Ethernet1/12*.
`state`	The status of PFC on the switch; one of: *off (disabled)*, *on (enabled)*.

Get PFC Interface Statistics

Gets PFC statistics for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/cee/pfc/counters/{*if_name*}
Request Body (JSON)	

Response

Response Body (JSON)	{
	"if_name":"{*if_name*}",
	"pfc_received":{*Rx count*},
	"pfc_sent":{*Tx count*}
	}

where:

Element	Description
`if_name`	The name of the switch interface, for example: *Ethernet1/12*.
`pfc_received`	The number of received PFC packets (integer).
`pfc_sent`	The number of sent PFC packets (integer).

Get ETS Configuration

Gets the current Enhanced Transmission Selection (ETS) configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/cee/ets
Request Body (JSON)	

Response

Response Body (JSON)

```
[
   {
      "pgid": 0,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 1,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 2,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 3,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 4,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 5,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 6,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 7,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 15,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
]
```

where:

Element	Description
`pgid`	The ID of the ETS priority group; an integer from 0-7, or 15.

Element	Description
`bandwidth`	The bandwidth percentage allocated to the priority group; an integer from 0-100.
`priority_ pgid_ mapping`	The priorities mapped to the priority group; an integer from 0-7.

Update ETS Configuration

Updates the ETS configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/cee/ets
Request Body (JSON)	[{ "pgid": 0, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 1, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 2, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 3, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 4, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 5, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 6, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 7, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 15, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" },]

where:

Element	Description
`pgid`	The ID of the ETS priority group; an integer from 0-7, or 15.
`bandwidth`	The bandwidth percentage allocated to the priority group; an integer from 0-100. **Note:** The total bandwidth percentage for all priority groups must not exceed 100%
`priority_ pgid_ mapping`	The priorities mapped to the priority group; an integer from 0-7.

Response

Response Body (JSON)

```
[
   {
      "pgid": 0,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 1,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 2,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 3,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 4,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 5,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 6,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 7,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
   {
      "pgid": 15,
      "bandwidth":"{banwidth}",
      "priority_pgid_mapping":"{priority_map}"
   },
]
```

where:

Element	Description
`pgid`	The ID of the ETS priority group; an integer from 0-7, or 15.

Element	Description
`bandwidth`	The bandwidth percentage allocated to the priority group; an integer from 0-100.
`priority_ pgid_ mapping`	The priorities mapped to the priority group; an integer from 0-7.

Get Application Protocol Configuration

Gets the current application protocol configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/cee/app-proto
Request Body (JSON)	

Response

Response Body (JSON)	[{ "config_name":"{*config_name*}", "protocol":"{*protocol*}", "protoid":"{*proto_id*}", "priority":"{*priority*}" },]

where:

Element	Description
`config_name`	The name of application protocol configuration (string).
`protocol`	The name of the protocol (string). For example: Ethertype, TCP, or UDP.
`protoid`	The protocol identifier; one of: • *hexadecimal* for Ethernet type (for example, *0x8900*) • `fcoe`, `iscsi`, `roce`, or `rocev2` for well known protocols • *0-65535* for TCP or UDP ports
`priority`	The priority mapped to the application protocol configuration; an integer from 0-7.

Create Application Protocol Configuration

Creates a new application protocol configuration.

Request

Method Type	POST
Request URI	/nos/api/cfg/cee/app-proto
Request Body (JSON)	[{ "config_name":"{*config_name*}", "protocol":"{*protocol*}", "protoid":"{*proto_id*}", "priority":"{*priority*}" },]

where:

Element	Description
`config_name`	The name of application protocol configuration (string).
`protocol`	(Optional) The unique name of the configured application protocol (string). For exammple: Ethertype, TCP, or UDP. **Note:** For well known protocols (fcoe, iscisi, roce, rocev2) this variable is not mandatory. For user defined protocols, the value of the variable must be *ethertype* and `protoid` must be a valid Ethernet type value (*0x800 - 0x9100*).

`protoid`	The protocol identifier; one of:
	• *hexadecimal* for Ethernet type (for example, *0x8900*)
	• `fcoe`, `iscsi`, `roce`, or `rocev2` for well known protocols
	• *0-65535* for TCP or UDP ports
	Note: This variable is optional for well known protocols
	To create a new application protocol for other proto-IDs, protocol information is required as showed in the following example for TCP port 8090.
	`Protocol  ProtoId  Priorities  ConfigName`
	`==========================================`
	`Ethertype  fcoe  5  fcoe_app`
	`TCP  iSCSI  4  iscsi_app`
	`UDP  RoCEv2  3  rocev2_app`
	`Ethertype  0x8900  1  my_eth_app`

`priority`	The priority mapped to the application protocol configuration; an integer from 0-7.

Response

Response Body (JSON)	[{ "config_name":"{*config_name*}", "protocol":"{*protocol*}", "protoid":"{*proto_id*}", "priority":"{*priority*}" },]

where:

Element	Description
`config_name`	The name of application protocol configuration (string).
`protocol`	The name of the protocol (string). For example: Ethertype, TCP, or UDP.
`protoid`	The protocol identifier; one of: • *hexadecimal* for Ethernet type (for example, *0x8900*) • `fcoe`, `iscsi`, `roce`, or `rocev2` for well known protocols • *0-65535* for TCP or UDP ports
`priority`	The priority mapped to the application protocol configuration; an integer from 0-7.

Delete Application Protocol Configuration

Deletes an application protocol configuration.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/cee/app-proto/{*config_name*}
Request Body (JSON)	

where:

Element	Description
config_name	The name of application protocol configuration (string).

Get DCBX Configuration

Gets the current Data Center Bridging Exchange (DCBX) configuration for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/cee/dcbx/{*if_name*}
Request Body (JSON)	

Response

Response Body (JSON)	{
	"if_name":"{*if_name*}",
	"dcbx state":"{enable|disable}",
	"pfc advt":"{on|off}",
	"ets advt":"{on|off}",
	"app advt":"{on|off}"
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`dcbx state`	The status of DCBX on the interface; one of: *enable, disable.*
`pfc advt`	The status of PFC local configuration advertisement to the DCBX peer; one of: *on, off.*
`ets advt`	The status of ETS local configuration advertisement to the DCBX peer; one of: *on, off.*
`app advt`	The status of application protocol local configuration advertisement to the DCBX peer; one of: *on, off.*

Update DCBX Configuration

Updates the DCBX configuration for a specific switch interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/cee/dcbx/{*if_name*}

Request Body (JSON)	{
	"if_name":"{*if_name*}",
	"dcbx state":"{enable|disable}",
	"pfc advt":"{on|off}",
	"ets advt":"{on|off}",
	"app advt":"{on|off}"
	}

where:

Element	Description
`if_name`	The name of the switch interface.For example: *Ethernet1/12*.
`dcbx state`	The status of DCBX on the interface; one of *enable, disable.* Default value: *on*
`pfc advt`	The status of PFC local configuration advertisement to the DCBX peer; one of *on, off.* Default value: *on*
`ets advt`	The status of ETS local configuration advertisement to the DCBX peer; one of: *on, off.* Default value: *on*
`app advt`	The status of application protocol local configuration advertisement to the DCBX peer; one of *on, off.* Default value: *on*

Response

Response Body (JSON)	{
	"if_name":"{*if_name*}",
	"dcbx state":"{enable|disable}",
	"pfc advt":"{on|off}",
	"ets advt":"{on|off}",
	"app advt":"{on|off}"
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`dcbx state`	The status of DCBX on the interface; one of *enable, disable*.
`pfc advt`	The status of PFC local configuration advertisement to the DCBX peer; one of *on, off*.

Element	Description
`ets advt`	The status of ETS local configuration advertisement to the DCBX peer; one of *on, off*.
`app advt`	The status of application protocol local configuration advertisement to the DCBX peer; one of *on, off*.

Get DCBX Control Interface Information

Gets DCBX control information for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/cee/dcbx/ctrl/{*if_name*}
Request Body (JSON)	

Response

Response Body (JSON)	{
	"if_name":"{*if_name*}",
	"DCBX Admin-state":"{disable|enable}",
	"DCBX version":"{*dcbx_version*}"
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`DCBX Admin-state`	The status of DCBX on the switch interface; one of *enable, disable*.
`DCBX version`	The version of DCBX; one of: • *DCBX CEE (v1.01)* • *DCBX IEEE 802.1Qaz (v2.5)*
`seq_no`	The sequence number, in case of DCBX CEE.
`ack_no`	The acknowledgement number, in case of DCBX CEE.

Get DCBX Administrative Interface Information

Gets DCBX administrative information for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/cee/dcbx/admin/{*if_name*}
Request Body (JSON)	

Response

Response Body (JSON)	<pre>{
	"if_name":"{*if_name*}",
	"dcbx_state":"{enable|disable}",
	"pfc_admin":
	{
	"state":"{on|off}",
	"advt":"{on|off}",
	"willing":"{on|off}",
	"max_cap":"{*max_cap*}",
	"syncd":"{on|off}",
	"priority_map":"{*priority_map*}"
	},
	"ets_admin":
	{
	"state":"{on|off}",
	"advt":"{on|off}",
	"willing":"{on|off}",
	"syncd":"{on|off}",
	"tcg":
	[
	{
	"pgid": 0,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 1,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 2,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 3,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	...</pre>

Response Body (JSON)	...
	{
	"pgid": 4,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 5,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 6,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 7,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 15,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	]
	},
	"app_admin":
	{
	"state":"{on|off}",
	"advt":"{on|off}",
	"willing":"{on|off}",
	},
	"app_protocol":
	[
	{
	"protocol":"{*protocol*}",
	"protostr":"{*proto_str*}",
	"priority":"{*priority*}"
	},
	]
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`dcbx state`	The status of DCBX on the interface; one of *enable, disable*.
`pfc_admin`	The current PFC configuration.
`state`	The status of PFC on the switch interface; one of *on, off*.
`advt`	The status of PFC local configuration advertisement to the DCBX peer; one of *on, off*.

Element	Description
`willing`	Whether the switch is "willing" to learn PFC configurations from a DCBX peer; one of *on, off*.
`max_cap`	The maximum PFC capability (integer).
`syncd`	The status of PFC information synchronization; one of *on, off*.
`priority_map`	The PFC priorities enabled on the interface; an integer from 0-7.
`ets_admin`	The current ETS configuration.
`state`	The status of ETS on the switch interface; one of *on, off*.
`advt`	The status of ETS local configuration advertisement to the DCBX peer; one of *on, off*.
`willing`	Whether the switch is "willing" to learn ETS configurations from a DCBX peer; one of *on, off*.
`syncd`	The status of ETS information synchronization; one of *on, off*.
`pgid`	The ID of the ETS priority group; an integer from *0-7*, or *15*.
`bandwidth`	The bandwidth percentage allocated to the priority group; an integer from *0-100*.
`priority_pgid_mapping`	The priorities mapped to the priority group; an integer from *0-7*.
`app_admin`	The current application control configuration.
`state`	The status of application control on the switch interface; one of *on, off*.
`advt`	The status of application control local configuration advertisement to the DCBX peer; one of *on, off*.
`willing`	Whether the switch is "willing" to learn application control configurations from a DCBX peer; one of *on, off*.
`app_protocol`	The list of created application protocols (string).
`protocol`	The name of the protocol (string). For example: Ethertype, TCP, or UDP.
`protostr`	The protocol identifier; one of: • *string* for Ethertype or well known protocols • *0-65535* for TCP or UDP ports
`priority`	The priority mapped to the application protocol configuration; an integer from *0-7*.

Get DCBX Operational Interface Information

Gets DCBX operational information for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/cee/dcbx/oper/{*if_name*}
Request Body (JSON)	

Response

Response Body (JSON)

```
{
    "if_name":"{if_name}",
    "dcbx_state":"{enable|disable}",
    "pfc_admin":
      {
         "state":"{on|off}",
         "max_cap":"{max_cap}",
         "syncd":"{on|off}",
         "priority_map":"{priority_map}"
      },
    "ets_admin":
      {
         "state":"{on|off}",
         "tcg":
           [
             {
                "pgid": 0,
                "bandwidth":"{banwidth}",
                "priority_pgid_mapping":"{priority_map}"
             },
             {
                "pgid": 1,
                "bandwidth":"{banwidth}",
                "priority_pgid_mapping":"{priority_map}"
             },
             {
                "pgid": 2,
                "bandwidth":"{banwidth}",
                "priority_pgid_mapping":"{priority_map}"
             },
             {
                "pgid": 3,
                "bandwidth":"{banwidth}",
                "priority_pgid_mapping":"{priority_map}"
             },
             {
                "pgid": 4,
                "bandwidth":"{banwidth}",
                "priority_pgid_mapping":"{priority_map}"
             },
...
```

Response Body (JSON)	...
	{
	"pgid": 5,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 6,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 7,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 15,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	]
	},
	"app_admin":
	{
	"state":"{on|off}",
	},
	"app_protocol":
	[
	{
	"protocol":"{*protocol*}",
	"protostr":"{*proto_str*}",
	"priority":"{*priority*}"
	},
	]
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`dcbx state`	The status of DCBX on the interface; one of *enable, disable*.
`pfc_admin`	The current PFC configuration.
`state`	The status of PFC on the switch interface; one of *on, off*.
`max_cap`	The maximum PFC capability (integer).
`syncd`	The status of PFC information synchronization; one of *on, off*.
`priority_map`	The PFC priorities enabled on the interface; an integer from *0-7*.
`ets_admin`	The current ETS configuration.
`state`	The status of ETS on the switch interface; one of *on, off*.

Element	Description
`pgid`	The ID of the ETS priority group; an integer from *0-7*, or *15*.
`bandwidth`	The bandwidth percentage allocated to the priority group; an integer from *0-100*.
`priority_ pgid_ mapping`	The priorities mapped to the priority group; an integer from *0-7*.
`app_admin`	The current application control configuration.
`state`	The status of application control on the switch interface; one of *on, off*.
`app_ protocol`	The list of created application protocols (string).
`protocol`	The name of the protocol. For example: Ethertype, TCP, or UDP.
`protostr`	The protocol identifier; one of: • *string* for Ethertype or well known protocols • *0-65535* for TCP or UDP ports
`priority`	The priority mapped to the application protocol configuration; an integer from *0-7*.

Get DCBX Remote Interface Information

Gets DCBX remote information for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/cee/dcbx/remote/{*if_name*}
Request Body (JSON)	

Response

Response Body (JSON)	{ "if_name":"{*if_name*}", "dcbx_state":"{enable|disable}", "pfc_admin": { "state":"{on|off}", "willing":"{on|off}", "max_cap":"{*max_cap*}", "priority_map":"{*priority_map*}" }, "ets_admin": { "state":"{on|off}", "willing":"{on|off}", "tcg": [{ "pgid": 0, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 1, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 2, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 3, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, { "pgid": 4, "bandwidth":"{*banwidth*}", "priority_pgid_mapping":"{*priority_map*}" }, ...

Response Body (JSON)	...
	{
	"pgid": 5,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 6,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 7,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	{
	"pgid": 15,
	"bandwidth":"{*banwidth*}",
	"priority_pgid_mapping":"{*priority_map*}"
	},
	]
	},
	"app_admin":
	{
	"state":"{on|off}",
	"willing":"{on|off}",
	},
	"app_protocol":
	[
	{
	"protocol":"{*protocol*}",
	"protostr":"{*proto_id*}",
	"priority":"{*priority*}"
	},
	]
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example *Ethernet1/12*.
`dcbx state`	The status of DCBX on the interface; one of *enable, disable*.
`pfc_admin`	The current PFC configuration.
`state`	The status of PFC on the switch interface; one of *on, off*.
`willing`	Whether the switch is "willing" to learn PFC configurations from a DCBX peer; one of *on, off*.
`max_cap`	The maximum PFC capability (integer).
`priority_map`	The PFC priorities enabled on the interface; an integer from *0-7*.
`ets_admin`	The current ETS configuration.

Element	Description
`state`	The status of ETS on the switch interface; one of *on, off*.
`willing`	Whether the switch is "willing" to learn ETS configurations from a DCBX peer; one of *on, off.*
`pgid`	The ID of the ETS priority group; an integer from *0-7*, or *15*.
`bandwidth`	The bandwidth percentage allocated to the priority group; an integer from *0-100*.
`priority_ pgid_ mapping`	The priorities mapped to the priority group; an integer from *0-7*.
`app_admin`	The current application control configuration.
`state`	The status of application control on the switch interface; one of *on, off*.
`willing`	Whether the switch is "willing" to learn application control configurations from a DCBX peer; one of *on, off*.
`app_ protocol`	The list of created application protocols (string).
`protocol`	The name of the protocol (string). For example Ethertype, TCP, or UDP.
`protostr`	The protocol identifier; one of: • *string* for Ethertype or well known protocols • *0-65535* for TCP or UDP ports
`priority`	The priority mapped to the application protocol configuration; an integer from *0-7*.

DCI

Note: Data Center Interconnection (DCI) is supported on the following Lenovo switches:

- RackSwitch G8272
- RackSwitch G8332
- RackSwitch G8296
- ThinkSystem NE1032T RackSwitch
- ThinkSystem NE1032 RackSwitch
- ThinkSystem NE1072T RackSwitch
- ThinkSystem NE10032 RackSwitch
- ThinkSystem NE2572 RackSwitch

The DCI URIs are available:

/nos/api/cfg/nwv	GET, POST
/nos/api/cfg/nwv/vxlan	GET, POST, PUT
/nos/api/cfg/<*if_name*>/vxlan	GET, POST, PUT
/nos/api/cfg/lag/<*id*>/vxlan	GET, POST, PUT
/nos/api/info/nwv/vxlan	GET
/nos/api/info/nwv/vxlan/mac-address	GET
/nos/api/info/nwv/vxlan/vni	GET
/nos/api/info/nwv/vxlan/vni/counters	GET, DELETE
/nos/api/info/nwv/vxlan/tunnel	GET
/nos/api/info/nwv/vxlan/virtual-port	GET
/nos/api/info/nwv/vxlan/virtual-port/counters	GET

The following DCI commands are available:

- Get NWV Configuration
- Update NWV Configuration
- Get NWV VXLAN (DCI) Configuration
- Set NWV VXLAN (DCI) Configuration
- Update NWV VXLAN (DCI) Configuration
- Get Interface Ethernet VXLAN Configuration
- Set Interface Ethernet VXLAN Configuration
- Update Interface Ethernet VXLAN Configuration
- Get Interface LAG VXLAN Configuration
- Set Interface LAG VXLAN Configuration
- Update Interface LAG VXLAN Configuration

- Get NWV VXLAN Information
- Get NWV VXLAN VNI Information
- Get NWV VXLAN VNI Counters Information
- Delete NWV VXLAN VNI Counters
- Get NWV VXLAN Tunnel Information
- Get NWV VXLAN MAC-Address Information
- Get NWV VXLAN Virtual Ports Information
- Get NWV VXLAN Virtual Port Counters Information

Get NWV Configuration

Gets global Network Virtualization (NWV) configuration of the system.

Request

Method Type	GET
Request URI	/nos/api/cfg/nwv
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	`{`
	`  "mode": "disabled | static | bgp-evpn ",`
	`  "ha"  : "disabled | vlag",`
	`  "encapsulation"  : "vtep"`
	`}`

where:

Element	Description
`mode`	NWM mode; one of *disabled, static, bgp-evpn*. Default value: *disabled*. **Note:** If the hardware install/uninstall process is not finished, “Vxlan not ready” is displayed.
`ha`	High Availability; one of *vlag*, *disabled*.
`encapsulation`	The Encapsulation method: `VTEP(VXLAN)`. **Note:** This parameter is valid for *hsc* and *disabled* modes.

Update NWV Configuration

Updates the NWV status.

Request

Method Type	POST
Request URI	/nos/api/cfg/nwv

Request Body
 	
(JSON)
	<pre>{
	"mode": "static",
	"ha" : "vlag",
	"encapsulation": "vtep"
	}</pre>

where:

Element	Description
`mode`	NWM mode; one of *disabled, static, bgp-evpn*. Default value: *disabled*. **Note:** If the hardware install/uninstall process is not finished, "Vxlan not ready" is displayed.
`ha`	High Availability; one of *vlag*, *disabled*.
`encapsulation`	The Encapsulation method: `VTEP(VXLAN)`. **Note:** This parameter is valid for *hsc* and *disabled* modes.

Response

Response Body
 	
(JSON)
	<pre>{
	"mode": "disabled | static | bgp-evpn ",
	"ha" : "disabled | vlag",
	"encapsulation" : "vtep"
	}</pre>

where:

Element	Description
`mode`	NWM mode; one of *disabled, static, bgp-evpn*. Default value: *disabled*. **Note:** If the hardware install/uninstall process is not finished, "Vxlan not ready" is displayed.
`ha`	High Availability; one of *vlag*, *disabled*.
`encapsulation`	The Encapsulation method: `VTEP(VXLAN)`. **Note:** This parameter is valid for *hsc* and *disabled* modes.

Get NWV VXLAN (DCI) Configuration

Gets NWV VXLAN (DCI) configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/nwv/vxlan
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"tunnel_ip_addr": "2.100.30.30",
	"vlan_bindings":[“map”
	[[101,10001],[201,20001]]
	],
	"remote_vtep": [“map”,
	[[10001,
	["10.10.1.1","10.2.2.3"]],
	[20001,
	["10.10.1.5"]
	]]
	]
	}

where:

Element	Description
`tunnel_ip_addr`	Configures the local VTEP interface.
`vlan_bindings`	Maps the global VLAN ID to a Virtual Network ID.
`remote_vtep`	Maps the VTEP to a specific VNID network.

Set NWV VXLAN (DCI) Configuration

Updates the NWV VXLAN (DCI) configuration.

Request

Method Type	POST
Request URI	/nos/api/cfg/nwv/vxlan

Request Body
 	
(JSON)
	<pre>{
	    "tunnel_ip_addr": "2.100.30.30",
	    "vlan_bindings": ["map", [[101,10001],[201,20001]]],
	    "remote_vtep": ["map",
	                    [[10001,
	                      ["10.10.1.1","10.2.2.3"]
	                    ],
	                    [20001,
	                      ["10.10.1.5"]
	                    ]
	                    ]
	                  ]
	}</pre>

where:

Element	Description
`tunnel_ip_addr`	(Mandatory) Configures the local VTEP interface; a valid IP address. **Note:** Make sure the IP address is properly configured.
`vlan_bindings`	(Optional) Configure the global VLAN-VNI mapping; a list of VLANID, VNID. The `map` parameter is used to add a new mapping to the existing one.
`remote_vtep`	(Optional) A list of VTEP IP address per VNID. The `map` parameter is used to add a new VTEP to the existing one.

Response

Response Body
 	
(JSON)
	<pre>{
	    "tunnel_ip_addr": "2.100.30.30",
	    "vlan_bindings": [“map,
	                      [[101,10001],
	                       [201,20001]]
	                      ],
	    "remote_vtep": [“map”,
	                     [[10001,
	                       ["10.10.1.1","10.2.2.3"]
	                      ],
	                      [20001,
	                       ["10.10.1.5"]
	                      ]
	                     ]
	                    ]
	}</pre>

where:

Element	Description
`tunnel_ip_addr`	Configures the local VTEP interface; a valid IP address. **Note:** Make sure the IP address is properly configured.
`vlan_bindings`	Configure the global VLAN-VNI mapping; a list of VLANID, VNID.
`remote_vtep`	A list of VTEP IP address per VNID. The `map` parameter is used to add a new VTEP to the existing one.

Update NWV VXLAN (DCI) Configuration

Updates the NWV VXLAN (DCI) configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/nwv/vxlan

Request Body
 	
(JSON)
	<pre>{
	    "tunnel_ip_addr": "2.100.30.30",
	    "vlan_bindings": ["set", [[201,20001]]],
	    "remote_vtep": ["set",
	                    [
	                    [10001,
	                      ["10.10.1.1","10.2.2.3"]
	                    ],
	                    [20001,
	                      ["10.10.1.5"]
	                    ]
	                    ]
	                  ]
	}</pre>

where:

Element	Description
`tunnel_ip_addr`	(Optional) Configures the local VTEP interface; a valid IP address. **Note:** Make sure the IP address is properly configured.
`vlan_bindings`	(Optional) Configure the global VLAN-VNI mapping; a list of VLANID, VNID. `set` removes the vni-mapping, `map` adds a new vni-mapping to the existing list.
`remote_vtep`	(Optional) A list of VTEP IP address per VNID. `set` removes the remote-vtep. The `map` parameter is used to add a new remote-vtep to the existing list.

Response

Response Body
 	
(JSON)
	{
	"tunnel_ip_addr": "2.100.30.30",
	"vlan_bindings": [“map”,
	[[101,10001], [201,20001]]
	],
	"remote_vtep": [“map”,
	[[10001,
	["10.10.1.1","10.2.2.3"]
	],
	[20001,
	["10.10.1.5"]
	]]
	],
	}

where:

Element	Description
`tunnel_ip_addr`	Configures the local VTEP interface; a valid IP address. **Note:** Make sure the IP address is properly configured.
`vlan_bindings`	Configure the global VLAN-VNI mapping; a list of VLANID, VNID.
`remote_vtep`	A list of VTEP IP address per VNID.

Get Interface Ethernet VXLAN Configuration

Gets VXLAN configuration for a specified interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/<*if_name*>/vxlan
Request Body (JSON)	

where:

Element	Description
if_name	Interface name. **Note:** The interface must exist. For example *Ethernet 1/1* or *Ethernet1%2F1* for RESTfulAPI.

Response

Response Body
 	
(JSON)
	<pre>{
	    "vxlan": "enabled | disabled"
	}</pre>

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist. For example *Ethernet 1/1* or *Ethernet1%2F1* for RESTfulAPI.
`vxlan`	The VXLAN interface state; one of *enabled, disabled*. Default value: *disabled*.

Set Interface Ethernet VXLAN Configuration

Updates the VXLAN configuration for a specified interface.

Request

Method Type	POST
Request URI	/nos/api/cfg/<*if_name*>/vxlan

Request Body
 	
(JSON)
	<pre>{
	  “vxlan”: “enabled | disabled |”
	}</pre>

where:

Element	Description
if_name	(Mandatory) Interface name. **Note:** The interface must exist. For example *Ethernet 1/1* or *Ethernet1%2F1* for RESTfulAPI.
`vxlan`	(Mandatory) The VXLAN interface state; one of *enabled, disabled*. Default value: *disabled* (null).

Response

Response Body
 	
(JSON)
	<pre>{
	  "vxlan": "disabled"
	}</pre>

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`vxlan`	The VXLAN interface state; one of *enabled, disabled*. Default value: *disabled*.

Update Interface Ethernet VXLAN Configuration

Updates the VXLAN configuration for a specified interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/<*if_name*>/vxlan

Request Body
(JSON)
	{
	"vxlan": "enabled | disabled"
	}

where:

Element	Description
if_name	(Mandatory) Interface name. **Note:** The interface must exist.
`vxlan`	(Mandatory) The VXLAN interface state; one of `enabled`, `disabled`. Default value: `disabled`.

Response

Response Body
(JSON)
	{
	"vxlan": "disabled"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`vxlan`	The VXLAN interface state; one of *enabled, disabled*. Default value: *disabled*.

Get Interface LAG VXLAN Configuration

Gets VXLAN configuration for a specified interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/lag/<*id*>/vxlan
Request Body (JSON)	

where:

Element	Description
id	LAG interface ID.

Response

Response Body
 	
(JSON)
	`{`
	`  "vxlan": "enabled | disabled"`
	`}`

where:

Element	Description
`id`	LAG interface ID.
`vxlan`	The VXLAN interface state; one of *enabled, disabled*. Default value: *disabled*.

Set Interface LAG VXLAN Configuration

Updates VXLAN configurations for a specified interface.

Request

Method Type	POST
Request URI	/nos/api/cfg/lag/<*id*>/vxlan

Request Body
 	
(JSON)
	{
	“vxlan”: “enabled | disabled”
	}

where:

Element	Description
id	(Mandatory) LAG interface ID.
`vxlan`	(Mandatory) The VXLAN interface state; one of `enabled`, `disabled`. Default value: `disabled` (null).

Response

Response Body
 	
(JSON)
	{
	"vxlan": "disabled"
	}

where:

Element	Description
`id`	LAG interface ID.
`vxlan`	The VXLAN interface state; one of *enabled, disabled*. Default value: *disabled*.

Update Interface LAG VXLAN Configuration

Updates the VXLAN configuration for a specific interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/lag/<*id*>/vxlan

Request Body
 	
(JSON)
	<pre>{
	  “vxlan”: “enabled | disabled”
	}</pre>

where:

Element	Description
id	LAG interface ID.
`vxlan`	The VXLAN interface state; one of `enabled`, `disabled`. Default value: `disabled`.

Response

Response Body
 	
(JSON)
	<pre>{
	  "vxlan": "disabled"
	}</pre>

where:

Element	Description
`id`	LAG interface ID.
`vxlan`	The VXLAN interface state; one of *enabled, disabled*. Default value: *disabled*.

Get NWV VXLAN Information

Gets the NWV VXLAN information.

Request

Method Type	GET
Request URI	/nos/api/info/nwv/vxlan
Request Body (JSON)	

Response

Response Body (JSON)	<pre>{
	    "tunnel_ip_addr": "2.100.30.30",
	    "vlan_bindings":[“map”,
	                    [[101,10001]]
	                  ],
	    "remote_vtep": [“map”,
	                    [[10001,
	                      ["10.10.1.1","10.2.2.3"]
	                    ],
	                    [20001,
	                      ["10.10.1.5"]
	                    ]]
	                  ]
	}</pre>

where:

Element	Description
`tunnel_ip_addr`	Configures the local VTEP interface; a valid IP address.
`vlan_bindings`	The global VLAN ID to Virtual Network ID mapping.
`remote_vtep`	The VTEP mapping to a specific VNID network.

Get NWV VXLAN VNI Information

Gets NWV VXLAN VNI information.

Request

Method Type	GET
Request URI	/nos/api/info/nwv/vxlan/vni
Request Body (JSON)	

Response

Response Body (JSON)	<pre>{
	“count”:2,
	“vni”: [10000,10001]
	}</pre>

where:

Element	Description
`count`	The total number of configured Virtual Network Instances (VNI).
`vni`	The list of configured VNI.

Get NWV VXLAN VNI Counters Information

Displays the NWV VXLAN virtual network counters.

Request

Method Type	GET
Request URI	/nos/api/info/nwv/vxlan/vni/counters
Request Body (JSON)	

Response

Response Body (JSON)	<pre>[{
	"bin": 583118262,
	"pktin": 695709,
	"bout": 1106560846,
	"pktout": 3099569,
	"vni": 1
	},
	{
	"bin": 768436428,
	"pktin": 929277,
	"bout": 1104301875,
	"pktout": 3044261,
	"vni": 2
	}]</pre>

where:

Element	Description
`bin`	Number of bytes received.
`pktin`	Number of packets in.
`bout`	Number of bytes sent.
`pktout`	Number of packets out.
`vni`	The number of configured VNI.

Delete NWV VXLAN VNI Counters

Resets the NWV VXLAN virtual network counters.

Request

Method Type	DELETE
Request URI	/nos/api/info/nwv/vxlan/vni/counters
Request Body (JSON)	

Response

Response Body (JSON)	

Get NWV VXLAN Tunnel Information

Gets NWV VXLAN tunnel information.

Request

Method Type	GET
Request URI	/nos/api/info/nwv/vxlan/tunnel
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	  “count”:1,
	“tunnel":[
	  {“remote-ip”:”192.168.1.3”, “status”:”UP”},
	  {“remote-ip”:”192.168.1.4”,”status”:”UP”},
	  {“remote-ip”:”192.168.1.4”,”status”:”UP”},
	  ]
	}</pre>

where:

Element	Description
`count`	The total number of tunnels.
`tunnel`	The list of configured VTEP. The remote IP address and BFD status are displayed for each tunnel.

Get NWV VXLAN MAC-Address Information

Gets NWV VXLAN MAC-Address information.

Request

Method Type	GET
Request URI	/nos/api/info/nwv/vxlan/mac-address
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{"local-count": 3048,
	"local-umac": [{"tunnel": "200.100.0.128", "mac":
	"00:B3:33:33:33:38", "vni": 206},
	{"tunnel": "200.100.0.128", "mac":
	"00:B3:33:33:33:39", "vni": 207},
	{"tunnel": "200.100.0.128", "mac":
	"00:B3:33:33:33:3A", "vni": 208},
	{"tunnel": "10.99.250.1", "mac":
	"00:AA:AA:AB:18:09", "vni": 3000},
	{"tunnel": "10.99.250.1", "mac":
	"00:AA:AA:AB:27:A9", "vni": 3000}]
	}</pre>

where:

Element	Description
`local-count`	Number of local MAC addresses.
`remote-count`	Number of remote MAC addresses.
`local-umac`	List of local MAC address. Each MAC address has an associated VNI and VTEP.
`remote-umac`	List of remote MAC address. Each MAC address has an associated VNI and VTEP.

Get NWV VXLAN Virtual Ports Information

Displays the virtual ports.

Request

Method Type	GET
Request URI	/nos/api/info/nwv/vxlan/virtualport
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	"count": 2,
	"virtual-port": [
	{"remoteTEP": "LOCAL", "vlan": "10",
	"port": "Ethernet1/7/2(A)", "vnid": "10001"},
	{"remoteTEP": "18.18.200.3", "vlan": "4093",
	"port": "Ethernet1/7/1(N,M)", "vnid": "10001"}]
	}</pre>

where:

Element	Description
`count`	The total number of virtual ports.
`virtual-port`	The list of virtual-ports and associated interface name, VNI and VLAN. The type is displayed for each virtual port (local/remote) VLAN and VNI.

Get NWV VXLAN Virtual Port Counters Information

Displays the virtual ports counters.

Request

Method Type	GET
Request URI	/nos/api/info/nwv/vxlan/virtualport/counters
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	“virtualport-counters”:
	[{"bin": 0, "vnid": "201", "pktout": 0, "pktin": 0, "bout":
	0, "port": "po100"},
	{"bin": 2343960, "vnid": "201", "pktout": 67404, "pktin":
	34470, "bout": 2239512, "port": "po4000"},
	{"bin": 0, "vnid": "202", "pktout": 0, "pktin": 0, "bout":
	0, "port": "po100"},
	{"bin": 2343960, "vnid": "202", "pktout": 39712, "pktin":
	34470, "bout": 356456, "port": "po4000"},
	
	{"bin": 54930725, "vnid": "2999", "pktout": 115577,
	"pktin": 68947, "bout": 52406893, "port": "po520"},
	{"bin": 0, "vnid": "3000", "pktout": 0, "pktin": 0, "bout":
	0, "port": "po100"},
	{"bin": 54951789, "vnid": "3000", "pktout": 115701,
	"pktin": 68947, "bout": 52511883, "port": "po520"},
	{"bin": 0, "vnid": "MULTIPLE", "pktout": 0, "pktin": 0,
	"bout": 0, "port": "Ethernet1/11"},
	{"bin": 0, "vnid": "MULTIPLE", "pktout": 0, "pktin": 0,
	"bout": 0, "port": "Ethernet1/11"}]</pre>

where:

Element	Description
`virtual-port-counters`	The list of counters for each virtual-port: • `bin`: number of bytes in • `bout`: number of bytes out • `pktout`: number of packets out • `pktin:` number of packets in

Default IP Address

The following default IP address URI is available:

- /nos/api/cfg/defaultipaddress/ GET, POST

The following default IP address commands are available:

- Get Default IP Address
- Set Default IP Address

Get Default IP Address

Gets default IP address on the management interface.

Request

Method Type Type	GET
Request URI	/nos/api/cfg/defaultipaddress/
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	`{`
	`  "interface": "mgmt0",`
	`  "state": <`*state_value*`>`
	`}`

where:

Element	Description
`interface`	The interface name; a string. **Note:** The interface must exist.
`state`	The state value; one of *set*, *unset*.

Set Default IP Address

Sets default IP address on the management interface.

Request

Method Type Type	POST
Request URI	/nos/api/cfg/defaultipaddress/

Request Body
(JSON)
	<pre>{
	"state": < state_value >
	}</pre>

where:

Element	Description
`state`	The state value; one of *set*, *unset*.

Response

Response Body
(JSON)
	<pre>{
	"interface": "mgmt0",
	"state": <*state_value*>
	}</pre>

where:

Element	Description
`interface`	The interface name; a string. **Note:** The interface must exist.
`state`	The state value; one of *set*, *unset*.

DHCP

The following DHCP URIs are available:

- /nos/api/cfg/dhcp GET, PUT
- /nos/api/cfg/dhcp_client/interface GET
- /nos/api/cfg/dhcp_client/interface/<*if_name*> PUT
- /nos/api/cfg/dhcp_client/interface/class_id/<*if_name*> DELETE
- /nos/api/cfg/dhcp_relay GET, PUT
- /nos/api/cfg/dhcp_relay/interface GET
- /nos/api/cfg/dhcp_relay/interface/<*if_name*> GET, PUT
- /nos/api/cfg/dhcp_relay/interface/<*if_name*>/<*relay_address*> DELETE
- /nos/api/cfg/dhcpsnp/entry GET, POST
- /nos/api/cfg/dhcpsnp/entry/<*mac_vlan_ifname*> DELETE
- /nos/api/cfg/dhcpsnp GET, PUT
- /nos/api/cfg/dhcpsnp/vlan GET
- /nos/api/cfg/dhcpsnp/vlan/<*vlan_id*> PUT, DELETE
- /nos/api/cfg/dhcpsnp/trust GET, PUT
- /nos/api/info/dhcpsnp/cnt GET, DELETE

The following DHCP commands are available:

- Get the Global DHCP Client Feature Property
- Update DHCP Client Feature Property
- Get DHCP Client Properties of All Interfaces
- Update DHCP Client Interface Properties
- Delete the Vendor Class Identifier of an Interface
- Get the Global DHCP Relay Service Property
- Update the Global DHCP Relay Service Property
- Get the DHCP Relay Properties of All Interfaces
- Get DHCP Relay Interface Properties
- Update DHCP Relay Interface Properties
- Delete DHCP Relay Interface Properties
- Get DHCP Snooping Binding Table Entry
- Create DHCP Snooping Binding Table Entry
- Delete DHCP Snooping Binding Table Entry
- Get DHCP Snooping Configuration

- Update DHCP Snooping Configuration
- Get DHCP Snooping VLAN Configuration
- Update DHCP Snooping VLAN Configuration
- Delete DHCP Snooping VLAN Configuration
- Get DHCP Snooping Trusted Ports Configuration
- Update DHCP Snooping Trusted Ports Configuration
- Get DHCP Snooping Statistics
- Clear DHCP Snooping Statistics

Get the Global DHCP Client Feature Property

Gets the global DHCP client feature property (whether or not DHCP is globally enabled).

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcp
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  "ena_dhcp_feature": "*<ena_dhcp_feature>*"
	}

where:

Element	Description
`ena_dhcp_feature`	Whether the DHCP client feature is enabled globally; one of *yes, no*. Default value: *yes*. **Note:** If disabled globally, DHCP client and DHCP relay is disabled on all interfaces. If enabled globally, the per-interface setting of DHCP client and DHCP relay takes effect.

Update DHCP Client Feature Property

Sets the global DHCP client feature property (whether or not DHCP is globally enabled).

Request

Method Type	PUT
Request URI	/nos/api/cfg/dhcp

Request Body
(JSON)
	{
	"ena_dhcp_feature": "*<ena_dhcp_feature>*"
	}

where:

Element	Description
`ena_dhcp_feature`	Whether the DHCP client feature is enabled globally; one of *yes, no*. Default value: *yes*. **Note:** If disabled globally, DHCP client and DHCP relay is disabled on all interfaces. If enabled globally, the per-interface setting of DHCP client and DHCP relay takes effect.

Response

Response Body
(JSON)
	{
	"ena_dhcp_feature": "*<ena_dhcp_feature>*"
	}

where:

Element	Description
`ena_dhcp_feature`	Whether the DHCP client feature is enabled globally; one of *yes, no*. Default value: *yes*. **Note:** If disabled globally, DHCP client and DHCP relay is disabled on all interfaces. If enabled globally, the per-interface setting of DHCP client and DHCP relay takes effect.

Get DHCP Client Properties of All Interfaces

Gets DHCP client properties of all interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcp_client/interface/
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"if_name": "*<if_name>*",
	"ena_v4_client": "*<ena_v4_client>*",
	"ena_v6_client": "*<ena_v6_client>*",
	"req_hostname": "*<req_hostname>*",
	"req_ntp_server": "*<req_ntp_server>*",
	"req_log_server": "*<req_log_server>*",
	"class_id": "*<class_id>*"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`ena_v4_client`	Whether the DHCPv4 client is enabled on the interface; one of *Yes, No*. Default values: *Yes* for the management interface and *No* for other switch interfaces.
`ena_v6_client`	Whether the DHCPv6 client is enabled on the interface; one of *Yes, No*. Default values: *Yes* for the management interface and *No* for other switch interfaces.
`req_hostname`	Whether a request has been issued for the host name option on an interface; one of *Yes, No*. Default value: *No*.
`req_ntp_server`	Whether a request has been issued for the NTP-server option on an interface; one of *Yes, No*. Default value: *No*.
`req_log_server`	Whether a request has been issued for the Log server option on an interface; one of *Yes, No*. Default value: *No*.
`class_id`	The name of Vendor class identifier. **Note:** The Vendor class identifier name must exist.

Update DHCP Client Interface Properties

Updates DHCP client properties of a specific interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/dhcp_client/interface/<*if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"ena_v4_client": "<*ena_v4_client*>",
	"ena_v6_client": "<*ena_v6_client*>",
	"req_hostname": "<*req_hostname*>",
	"req_ntp_server": "<*req_ntp_server*>",
	"req_log_server": "<*req_log_server*>",
	"class_id": "<*class_id*>"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`ena_v4_client`	Whether the DHCPv4 client is enabled on the interface; one of *Yes, No*. Default values: *Yes* for the management interface and *No* for other switch interfaces.
`ena_v6_client`	Whether the DHCPv6 client is enabled on the interface; one of *Yes, No*. Default values: *Yes* for the management interface and *No* for other switch interfaces.
`req_hostname`	Whether a request has been issued for the host name option on an interface; one of *Yes, No*. Default value: *No*.
`req_ntp_server`	Whether a request has been issued for the NTP-server option on an interface; one of *Yes, No*. Default value: *No*.
`req_log_server`	Whether a request has been issued for the Log server option on an interface; one of *Yes, No*. Default value: *No*.
`class_id`	The name of Vendor class identifier. **Note:** The Vendor class identifier name must exist.

Response

Response Body
 	
(JSON)
	{
	"if_name": "*<if_name>*",
	"ena_v4_client": "*<ena_v4_client>*",
	"ena_v6_client": "*<ena_v6_client>*",
	"req_hostname": "*<req_hostname>*",
	"req_ntp_server": "*<req_ntp_server>*",
	"req_log_server": "*<req_log_server>*",
	"class_id": "*<class_id>*"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`ena_v4_client`	Whether the DHCPv4 client is enabled on the interface; one of *Yes, No*. Default values: *Yes* for the management interface and *No* for other switch interfaces.
`ena_v6_client`	Whether the DHCPv6 client is enabled on the interface; one of *Yes, No*. Default values: *Yes* for the management interface and *No* for other switch interfaces.
`req_hostname`	Whether a request has been issued for the host name option on an interface; one of *Yes, No*. Default value: *No*.
`req_ntp_server`	Whether a request has been issued for the NTP-server option on an interface; one of *Yes, No*. Default value: *No*.
`req_log_server`	Whether a request has been issued for the Log server option on an interface; one of *Yes, No*. Default value: *No*.
`class_id`	The name of Vendor class identifier. **Note:** The Vendor class identifier name must exist.

Delete the Vendor Class Identifier of an Interface

Deletes the vendor class identifier of a specific interface.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/dhcp_client/interface/class_id/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	Interface name. **Note:** The interface must exist.

Response

Response Body (JSON)	

Get the Global DHCP Relay Service Property

Gets the global DHCP relay service property (whether the relay service has been enabled globally) class identifier of a specific interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcp_relay
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"ena_v4_relay": "*<ena_v4_relay>*",
	"ena_v6_relay": "*<ena_v6_relay*",
	}

where:

Element	Description
`ena_v4_relay`	Whether DHCPv4 relay is enabled on the interface; one of *Yes, No*. Default value: *Yes*. **Note:** If disabled globally, DHCPv4 relay is disabled on all interfaces. If DHCPv4 relay service is enabled globally, the per-interface setting of DHCPv4 relay takes effect.
`ena_v6_relay`	Whether DHCPv6 relay is enabled on the interface; one of *Yes, No*. Default value: *Yes*. **Note:** If disabled globally, DHCPv6 relay is disabled on all interfaces. If DHCPv6 relay service is enabled globally, the per-interface setting of DHCPv6 relay takes effect.

Update the Global DHCP Relay Service Property

Updates the global DHCP relay service property (whether the relay service has been enabled globally) class identifier of a specific interfaces.

Request

Method Type	PUT
Request URI	/nos/api/cfg/dhcp_relay
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"ena_v4_relay": "*<ena_v4_relay>*",
	"ena_v6_relay": "*<ena_v6_relay*",
	}

where:

Element	Description
`ena_v4_relay`	Whether DHCPv4 relay is enabled on the interface; one of *Yes, No*. Default value: *Yes*. **Note:** If disabled globally, DHCPv4 relay is disabled on all interfaces. If DHCPv4 relay service is enabled globally, the per-interface setting of DHCPv4 relay takes effect.
`ena_v6_relay`	Whether DHCPv6 relay is enabled on the interface; one of *Yes, No*. Default value: *Yes*. **Note:** If disabled globally, DHCPv6 relay is disabled on all interfaces. If DHCPv6 relay service is enabled globally, the per-interface setting of DHCPv6 relay takes effect.

Get the DHCP Relay Properties of All Interfaces

Gets the DHCP relay properties for all interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcp_relay/interface/
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"if_name": "*<if_name>*",
	"dhcpv4_relay":
	[
	{
	"v4_relay_addr": "*<v4_relay_addr>*",
	}
	],
	"dhcpv6_relay":
	[
	{
	"v6_relay_addr": "*<v6_relay_addr>*",
	"v6_relay_out_if": "*<v6_relay_out_if>*"
	}
	]
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`v4_relay_addr`	IPv4 address of the relay server; a valid IPv4 address.
`v6_relay_addr`	IPv6 address of the relay server; a valid IPv6 address.
`v6_relay_out_if`	Outgoing interface of the relay service. **Note:** The interface must exist.

Get DHCP Relay Interface Properties

Gets the DHCP relay interface properties for a specific interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcp_relay/interface/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	Interface name. **Note:** The interface must exist.

Response

Response Body
 	
(JSON)
	<pre>[
	  {
	  "if_name": "<*if_name*>",
	  "dhcpv4_relay":
	  [
	  {
	  "v4_relay_addr": "<*v4_relay_addr*>",
	  }
	  ],
	  "dhcpv6_relay":
	  [
	  {
	  "v6_relay_addr": "<*v6_relay_addr*>",
	  "v6_relay_out_if": "<*v6_relay_out_if*>"
	  }
	  ]
	  }
	]</pre>

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`v4_relay_addr`	IPv4 address of the relay server; a valid IPv4 address.
`v6_relay_addr`	IPv6 address of the relay server; a valid IPv6 address.

Element	Description
`v6_relay_out_if`	Outgoing interface of the relay service. **Note:** The interface must exist.

Notes:

- The response body will be empty if no DHCP relay configuration has been made on the specified interface.
- Only the what is configured will be displayed. For example, if no `dhcpv6_relay` is configured, none will be displayed.

Update DHCP Relay Interface Properties

Updates the DHCP relay interface properties for a specific interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/dhcp_relay/interface/<*if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"dhcpv4_relay":
	[
	{
	"v4_relay_addr": "<*v4_relay_addr*>",
	}
	],
	"dhcpv6_relay":
	[
	{
	"v6_relay_addr": "<*v6_relay_addr*>",
	"v6_relay_out_if": "<*v6_relay_out_if*>"
	}
	]
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`v4_relay_addr`	IPv4 address of the relay server; a valid IPv4 address.
`v6_relay_addr`	IPv6 address of the relay server; a valid IPv6 address.
`v6_relay_out_if`	Outgoing interface of the relay service. **Note:** The interface must exist.

Response

Response Body
 	
(JSON)
	[
	  {
	  "if_name": "*<if_name>*",
	  "dhcpv4_relay":
	  [
	  {
	  "v4_relay_addr": "*<v4_relay_addr>*",
	  }
	  ],
	  "dhcpv6_relay":
	  [
	  {
	  "v6_relay_addr": "*<v6_relay_addr>*",
	  "v6_relay_out_if": "*<v6_relay_out_if>*"
	  }
	  ]
	  }
	]

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`v4_relay_addr`	IPv4 address of the relay server; a valid IPv4 address.
`v6_relay_addr`	IPv6 address of the relay server; a valid IPv6 address.
`v6_relay_out_if`	Outgoing interface of the relay service. **Note:** The interface must exist.

Delete DHCP Relay Interface Properties

Deletes the DHCP relay interface properties for a specific interface.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/dhcp_relay/interface/<*if_name*>/<*relay_address*>
Request Body (JSON)	

where:

Element	Description
if_name	Interface name. **Note:** The interface must exist.
relay_address	IPv4 or IPv6 address of the relay server; a valid IPv4 or IPv6 address. **Note:** If no relay address is specified, all relay server configuration information for this interface will be removed.

Response

Response Body (JSON)	

Get DHCP Snooping Binding Table Entry

Gets the DHCP Snooping binding table entries.

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcpsnp/entry
Request Body (JSON)	

Response

Response Body (JSON)	[
	  {
	  "mac":"{*mac address*}",
	  "ip_addr":"{*ip address*}",
	  "lease_time":"{*lease time in seconds*}",
	  "type":"{*dynamic/static*}",
	  "vlan":"{*vlan number*}",
	  "if_name":"{*if_name*}",
	  }
	]

where:

Element	Description
`mac`	The MAC address of the binding table entry, in the following format: *XX:XX:XX:XX:XX:XX.*
`ip_addr`	The IP address of the binding table entry.
`lease_time`	The lease time, in seconds, for the binding table entry; an integer from *1-4294967295.*
`type`	The type of the binding table entry; one of *dynamic, static.*
`vlan`	The VLAN of the binding table entry; an integer from *1-4093.*
`if_name`	The name of the switch interface for the binding table entry. For example *Ethernet1/12*.

Create DHCP Snooping Binding Table Entry

Adds a DHCP Snooping binding table entry.

Request

Method Type	POST
Request URI	/nos/api/cfg/dhcpsnp/entry

Request Body (JSON)	{
	"mac":"{*mac address*}",
	"ip_addr":"{*ip address*}",
	"lease_time":"{*lease time in seconds*}",
	"vlan":"{*vlan number*}",
	"if_name":"{*if_name*}",
	}

where:

Element	Description
`mac`	The MAC address of the binding table entry, in the following format: *XX:XX:XX:XX:XX:XX*.
`ip_addr`	The IP address of the binding table entry.
`lease_time`	The lease time, in seconds, for the binding table entry; an integer from *1-4294967295*.
`type`	The type of the binding table entry; one of *dynamic, static*.
`vlan`	The VLAN of the binding table entry; an integer from *1-4093*.

Delete DHCP Snooping Binding Table Entry

Removes an entry from the DHCP Snooping binding table.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/dhcpsnp/entry/<*mac_vlan_ifname*>
Request Body (JSON)	

where:

Element	Description
mac_vlan_ifname	The MAC address, VLAN, or interface name to identify the binding table entry; one of: • *MAC address in format XX:XX:XX:XX:XX:XX* • VLAN number: *1-4093* • *Interface name* (for example, *Ethernet1/12*)

Get DHCP Snooping Configuration

Gets the DHCP Snooping configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcpsnp
Request Body (JSON)	

Response

Response Body (JSON)	`{`
	`  "dhcpsnp_feature":"{enable/disable}",`
	`  "option_82":"{enable/disable}",`
	`}`

where:

Element	Description
`dhcpsnp_feature`	The global status of the DHCP service on the switch; one of *enable, disable.*
`option_82`	The status of DHCP Option 82; one of *enable, disable.*

Update DHCP Snooping Configuration

Updates the DHCP Snooping configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/dhcpsnp

Request Body (JSON)	`{`
	`  "dhcpsnp_feature":"{enable/disable}",`
	`  "option_82":"{enable/disable}",`
	`}`

Response

Response Body (JSON)	

where:

Element	Description
`dhcpsnp_feature`	The global status of the DHCP service on the switch; one of *enable, disable.*
`option_82`	The status of DHCP Option 82; one of *enable, disable.*

Get DHCP Snooping VLAN Configuration

Gets the DHCP Snooping VLAN configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcpsnp/vlan
Request Body (JSON)	

Response

Response Body (JSON)	<pre>{
	    "vlan_enabled":"{vlan_id}",
	}</pre>

where:

Element	Description
`vlan_ enabled`	The VLAN for DHCP Snooping; an integer from *1-4093.*

Update DHCP Snooping VLAN Configuration

Updates the DHCP Snooping VLAN configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/dhcpsnp/vlan

Request Body (JSON)	`{`
	`  "vlan_enabled":{vlan_id}`
	`}`

where:

Element	Description
`vlan_enabled`	The VLAN on which DHCP Snooping is enabled; an integer from *1-4093*.

Delete DHCP Snooping VLAN Configuration

Deletes the DHCP Snooping VLAN configuration.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/dhcpsnp/vlan/*<vlan_id>*
Request Body (JSON)	

where:

Element	Description
`vlan_id`	The VLAN for DHCP Snooping; an integer from *1-4093*.

Get DHCP Snooping Trusted Ports Configuration

Gets the DHCP Snooping trusted ports configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/dhcpsnp/trust
Request Body (JSON)	

Response

Response Body (JSON)	`[`
	`  {`
	`  "if_name":"{`*interface name*`}",`
	`  "trusted":"{`*yes/no*`}"`
	`  }`
	`]`

where:

Element	Description
`if_name`	The name of the switch interface to be configured as a DHCP Snooping trusted port. For example: *Ethernet1/12*.
`trusted`	Whether the specified switch interface is a trusted port; one of *yes, no*.

Update DHCP Snooping Trusted Ports Configuration

Updates the DHCP Snooping trusted ports configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/dhcpsnp/trust

Request Body (JSON)	{
	"if_name":"{*interface name*}",
	"trusted":"{*yes/no*}"
	}

where:

Element	Description
`if_name`	The name of the switch interface to be configured as a DHCP Snooping trusted port. For example: *Ethernet1/12*.
`trusted`	Whether the specified switch interface is a trusted port. one of: *yes, no.*

Get DHCP Snooping Statistics

Gets DHCP Snooping statistics.

Request

Method Type	GET
Request URI	/nos/api/info/dhcpsnp/cnt
Request Body (JSON)	

Response

Response Body (JSON)	{
	"rcv_req_pkts":"{*Received Request packets*}",
	"rcv_rep_pkts":"{*Received Reply packets*}",
	"drop_pkts":"{*Dropped packets*}"
	}

where:

Element	Description
`rcv_req_pkts`	The number of received DHCP request packets; an integer from *0-4294967295.*
`rcv_rep_pkts`	The number of received DHCP reply packets. an integer from *0-4294967295.*
`drop_pkts`	The number of dropped DHCP packets. an integer from *0-4294967295.*

Clear DHCP Snooping Statistics

Resets DHCP Snooping statistics.

Request

Method Type	DELETE
Request URI	/nos/api/info/dhcpsnp/cnt
Request Body (JSON)	

DNS

The following Domain Name System (DNS) URIs are available:

- /nos/api/info/dns GET
- /nos/api/cfg/dns POST
- /nos/api/cfg/dns/nameserver POST, DELETE
- /nos/api/cfg/dns/defaultdomain POST, DELETE
- /nos/api/cfg/dns/domain POST, DELETE
- /nos/api/cfg/dns/nametoip POST, DELETE

The following DNS commands are available:

- Get DNS Host Information
- Configure DNS Client Service
- Configure DNS Name Server
- Delete DNS Name Server
- Configure DNS Default Domain Name
- Delete DNS Default Domain Name
- Configure DNS Domain Name
- Delete DNS Domain Name
- Configure DNS Hostname to IP Address Mapping
- Delete DNS Hostname to IP Address Mapping

Get DNS Host Information

Gets DNS Client information, such as domain-lookup service state, default domain name, and additional domain names.

Request

Method Type	GET
Request URI	/nos/api/info/dns
Request Body (JSON)	

Response

Response Body (JSON)	{
	"domain_lookup":"enable/disable",
	"dynamic_domain":"Dynamic domain/No dynamic domain",
	"dynamic_nameserver":"Dynamic name server/Static name server",
	"domain_list":[{"domain_name":"*<domain>*"}],
	"nameserver_list":[{"address":"*<address>*"}],
	"nametoip_list":[{"name":"*<name>*", "address":"*<address>*"}]
	}

where:

Element	Description
`domain_lookup`	The status of the DNS service on the switch; one of *enabled, disabled.*
`dynamic_domain`	Whether the DNS Client domain name is dynamically learnt; one of *dynamic domain, no dynamic domain.*
`dynamic_nameserver`	Whether the DNS Client name server is dynamically learnt; one of *dynamic name server, static name server.*
`domain_list`	(Optional) The DNS Client configured domain name list.
`nameserver_list`	(Optional) The DNS Client configured name server list.
`nametoip_list`	(Optional) The DNS Client configured hostname to IP address mapping list. Valid value: the list of hostname to IP address mappings.

Configure DNS Client Service

Configures the status of the DNS Client service on the switch.

Request

Method Type	POST
Request URI	/nos/api/cfg/dns

Request Body (JSON)	<pre>{
	  "dns_client_status":"enabled/disabled"
	}</pre>

where:

Element	Description
`dns_client_status`	The status of the DNS Client service; one of *enabled, disabled.*

Response

Response Body (JSON)	<pre>{
	  "dns_client_status":"enabled/disabled"
	}</pre>

Configure DNS Name Server

Configures a DNS name server.

Request

Method Type	POST
Request URI	/nos/api/cfg/dns/nameserver

Request Body (JSON)	{
	"nameserver1":"*<server1_address>*",
	"nameserver2":"*<server2_address>*",
	"nameserver3":"*<server3_address>*",
	"vrf":"*<vrf_name>*"
	}

where:

Element	Description
`nameserver1`	The first name server address. Valid value: the IP address.
`nameserver2`	The second name server address. Valid value: the IP address.
`nameserver3`	The third name server address. Valid value: the IP address.
`vrf`	The VRF instance for the name server. Valid value: the VRF instance name.

Delete DNS Name Server

Removes a configured DNS name server.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/dns/nameserver/<*server_addr*>/<*vrf*>

where:

Element	Description
server_addr	The name server address. Valid value: the IP address.
vrf	The VRF instance for the name server. Valid value: the VRF instance name. Default value: *default*.

Configure DNS Default Domain Name

Configures the default DNS domain name.

Request

Method Type	POST
Request URI	/nos/api/cfg/dns/defaultdomain

Request Body (JSON)	{
	"domain_name":"*<default_domain_name>*",
	"vrf":"*<vrf_name>*"
	}

where:

Element	Description
`domain_name`	The name of the default DNS domain (string).
`vrf`	The VRF instance for the default DNS domain. Valid value: the VRF instance name. Default value: *default.*

Delete DNS Default Domain Name

Removes the default DNS domain name.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/dns/defaultdomain/<*domain_name*>/<*vrf*>

where:

Element	Description
domain_name	The name of the default DNS domain (string).
vrf	The VRF instance for the default DNS domain. Valid value: the VRF instance name. Default value: `default`.

Configure DNS Domain Name

Configures a DNS domain name.

Request

Method Type	POST
Request URI	/nos/api/cfg/dns/domain

Request Body (JSON)	{
	"domain_name":"*<domain_name>*",
	"vrf":"*<vrf_name>*"
	}

where:

Element	Description
`domain_name`	The name of the default DNS domain (string).
`vrf`	The VRF instance for the default DNS domain. Valid value: the VRF instance name. Default value: *default.*

Delete DNS Domain Name

Removes a configured DNS domain name.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/dns/domain/<*domain_name*>/<*vrf*>

where:

Element	Description
domain_name	The name of the default DNS domain (string).
vrf	The VRF instance for the default DNS domain. Valid value: the VRF instance name. Default value: *default*.

Configure DNS Hostname to IP Address Mapping

Configures a DNS server hostname to IP address mapping.

Request

Method Type	POST
Request URI	/nos/api/cfg/dns/nametoip

Request Body (JSON)	{
	"hostname":"*<hostname>*",
	"ip_addr1":"*<address1>*",
	"ip_addr2":"*<address2>*",
	"vrf":"*<vrf_name>*"
	}

where:

Element	Description
`hostname`	The hostname of the DNS server (string).
`ip_addr1`	The first IP address of the DNS server.
`ip_addr2`	The second IP address of the DNS server.
`vrf`	The VRF instance for the default DNS domain. Valid value: the VRF instance name. Default value: *default.*

Delete DNS Hostname to IP Address Mapping

Removes a configured DNS server hostname to IP address mapping.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/dns/nametoip/<*hostname*>/<*addr*>/<*vrf*>

where:

Element	Description
hostname	The hostname of the DNS server (string).
addr	The IP address of the DNS server.
vrf	The VRF instance for the default DNS domain. Valid value: the VFR instance name. Default value: `default`.

ECMP

The following Equal Cost Multiple Paths (ECMP) URIs are available:

- /nos/api/cfg/ip/ecmp/weight GET, PUT
- /nos/api/info/ip/ecmp/weight/ipv4/<*ipv4_address*> GET
- /nos/api/info/ip/ecmp/weight/ipv6/<*ipv6_address*> GET
- /nos/api/info/ip/ecmp/weight/interface/<*interface_name*> GET
- /nos/api/cfg/ip/ecmp/weight/ipv4 POST
- /nos/api/cfg/ip/ecmp/weight/ipv6 POST
- /nos/api/cfg/ip/ecmp/weight/interface POST

The following ECMP commands are available:

- Get Weighted ECMP Status
- Update Weighted ECMP Status
- Get IPv4 Next-hop ECMP Weight
- Configure IPv4 Next-hop ECMP Weight
- Get IPv6 Next-hop ECMP Weight
- Configure IPv6 Next-hop ECMP Weight
- Get Interface ECMP Weight
- Configure Interface ECMP Weight

Get Weighted ECMP Status

Gets the status of weighted ECMP on the switch.

Request

Method Type	GET
Request URI	/nos/api/cfg/ip/ecmp/weight
Request Body (JSON)	

Response

Response Body (JSON)	<pre>{
	  "weighted_ecmp_state":"{Enable|Disable}"
	}</pre>

where:

Element	Description
`weighted_ecmp_state`	The status of weighted ECMP; one of *enable, disable.*

Update Weighted ECMP Status

Updates the status of weighted ECMP on the switch.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ip/ecmp/weight
Request Body (JSON)	`{ "weighted_ecmp_state":"{enable|disable}" }`

where:

Element	Description
`weighted_ecmp_state`	The status of weighted ECMP; one of *enable, disable.*

Response

Response Body (JSON)	`{ "weighted_ecmp_state":"{enable|disable}" }`

where:

Element	Description
`weighted_ecmp_state`	The status of weighted ECMP; one of *enable, disable.*

Get IPv4 Next-hop ECMP Weight

Gets the ECMP weight of a specified IPv4 next-hop.

Request

Method Type	GET
Request URI	nos/api/info/ip/ecmp/weight/ipv4/<*ipv4_address*>
Request Body (JSON)	

Response

Response Body (JSON)	{
	"ipv4_nexthop_address":"{*IPv4 address*}",
	"ipv4_nexthop_weight":{*weight*}
	}

where:

Element	Description
`ipv4_nexthop_address`	The IPv4 address of the next-hop.
`ipv4_nexthop_weight`	The ECMP weight of the specified next-hop; an integer from *1-4*.

Configure IPv4 Next-hop ECMP Weight

Configures the ECMP weight of a specific IPv4 next-hop.

Request

Method Type	POST
Request URI	/nos/api/cfg/ip/ecmp/weight/ipv4

Request Body (JSON)	{
	"ipv4_nexthop_address":"{*IPv4 address*}",
	"ipv4_nexthop_weight":{*weight*}
	}

where:

Element	Description
`ipv4_nexthop_address`	The IPv4 address of the next-hop.
`ipv4_nexthop_weight`	The ECMP weight of the specified next-hop; an integer from *1-4*.

Response

Response Body (JSON)	{
	"ipv4_nexthop_address":"{*IPv4 address*}",
	"ipv4_nexthop_weight":{*weight*}
	}

where:

Element	Description
`ipv4_nexthop_address`	The IPv4 address of the next-hop.
`ipv4_nexthop_weight`	The ECMP weight of the specified next-hop; an integer from *1-4*.

Get IPv6 Next-hop ECMP Weight

Gets the ECMP weight of a specified IPv6 next-hop.

Request

Method Type	GET
Request URI	/nos/api/info/ip/ecmp/weight/ipv6/<*ipv6_address*>
Request Body (JSON)	

Response

Response Body (JSON)	{
	"ipv6_nexthop_address":"{*IPv6 address*}",
	"ipv6_nexthop_weight":{*weight*}
	}

where:

Element	Description
`ipv6_nexthop_address`	The IPv6 address of the next-hop.
`ipv6_nexthop_weight`	The ECMP weight of the specified next-hop; an integer from *1-4*.

Configure IPv6 Next-hop ECMP Weight

Configures the ECMP weight of a specific IPv6 next-hop.

Request

Method Type	POST
Request URI	/nos/api/cfg/ip/ecmp/weight/ipv6

Request Body (JSON)	{
	"ipv6_nexthop_address":"{*IPv6 address*}",
	"ipv6_nexthop_weight":{*weight*}
	}

where:

Element	Description
`ipv6_nexthop_address`	The IPv6 address of the next-hop.
`ipv6_nexthop_weight`	The ECMP weight of the specified next-hop; an integer from *1-4*.

Response

Response Body (JSON)	{
	"ipv6_nexthop_address":"{*IPv6 address*}",
	"ipv6_nexthop_weight":{*weight*}
	}

where:

Element	Description
`ipv6_nexthop_address`	The IPv6 address of the next-hop.
`ipv6_nexthop_weight`	The ECMP weight of the specified next-hop; an integer from *1-4*.

Get Interface ECMP Weight

Gets the ECMP weight of a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/ip/ecmp/weight/interface/<*interface name*>
Request Body (JSON)	

Response

Response Body (JSON)	{
	"interface_name":"{*interface name*}",
	"interface_weight":{*weight*}
	}

where:

Element	Description
`interface_name`	The name of the switch interface.
`interface_weight`	The ECMP weight of the specified switch interface. an integer from *1-4*.

Configure Interface ECMP Weight

Configures the ECMP weight of a specific switch interface.

Request

Method Type	POST
Request URI	/nos/api/cfg/ip/ecmp/weight/interface

Request Body (JSON)	{
	"interface_name":"{*interface name*}",
	"interface_weight":{*weight*}
	}

where:

Element	Description
`interface_name`	The name of the switch interface.
`interface_weight`	The ECMP weight of the specified switch interface; an integer from *1-4*.

Response

Response Body (JSON)	{
	"interface_name":"{*interface name*}",
	"interface_weight":{*weight*}
	}

where:

Element	Description
`interface_name`	The name of the switch interface.
`interface_weight`	The ECMP weight of the specified switch interface; an integer from *1-4*.

FDB

The following Forwarding Database (FDB) URIs are available:

- /nos/api/info/fdb/list POST
- /nos/api/info/fdb/count POST
- /nos/api/info/fdb/global GET
- /nos/api/info/fdb/interface/<*if_name*> GET
- /nos/api/cfg/fdb GET, POST, PUT
- /nos/api/cfg/fdb/global GET, PUT
- /nos/api/cfg/fdb/interface/<*if_name*> GET, PUT

The following FDB commands are available:

- Get List of MAC Addresses
- Get Number of MAC Addresses
- Get Global FDB Runtime Settings
- Get Global FDB Configured Settings
- Update Global FDB Settings
- Get MAC Address Learning Interface Runtime Setting
- Get MAC Address Learning Interface Configured Setting
- Update Interface MAC Address Learning Setting
- Get Static MAC Addresses
- Create Static MAC Address
- Delete MAC Address or Interface for Multicast MAC Address

Get List of MAC Addresses

Gets all MAC addresses that match the search criteria.

Request

Method Type	POST
Request URI	/nos/api/info/fdb/list

Request Body (JSON)	{
	"fdb_type":"{static|multicast|dynamic}",
	"mac_address":"{*mac_address*}",
	"interfaces":["{*if_name*}"],
	"vlan_id":"{*vlan_id*}"
	}

where:

Element	Description
`fdb_type`	The type of MAC address to filter on; one of: • *static* • *dynamic* • *multicast*
`mac_address`	The MAC address matching the criteria.
`interfaces`	The name of the switch interface to filter on. For example: *Ethernet1/12*.
`vlan_id`	The VLAN number to filter on; an integer from *1-4094.*

Response

Response Body (JSON)	{
	"address_table":
	[
	{
	"vlan_id":"{*vlan_id*}",
	"mac_address":"{*mac_address*}",
	"is_static":"{true|false}",
	"if_name":"{*if_name*}"
	}
	]
	}

where:

Element	Description
`address_table`	The MAC table entries.
`vlan_id`	The VLAN number for the MAC table entry; an integer from *1-4094.*
`mac_address`	The MAC address matching the search criteria.
`is_static`	Whether the MAC address is statically configured; one of *true, false.*
`if_name`	The name of the switch interface to filter on. For example: *Ethernet1/12*.

Get Number of MAC Addresses

Gets the total number of MAC addresses matching the search criteria.

Request

Method Type	POST
Request URI	/nos/api/info/fdb/count

Request Body (JSON)	{
	"fdb_type":"{static|multicast|dynamic}",
	"mac_address":"{*mac_address*}",
	"interfaces":["{*if_name*}"],
	"vlan_id":"{*vlan_id*}"
	}

where:

Element	Description
`fdb_type`	The type of MAC address to filter on; one of: • *static* • *dynamic* • *multicast*
`mac_address`	The MAC address matching the criteria.
`interfaces`	The name of the switch interface to filter on. For example *Ethernet1/12*.
`vlan_id`	The VLAN number to filter on; an integer from *1-4094.*

Response

Response Body (JSON)	{
	"dynamic_add_cnt":"{*dynamic_add_cnt*}",
	"static_add_cnt":"{*static_add_cnt*}",
	"multicast_add_cnt":"{*multicast_add_cnt*}",
	"total_in_use_cnt":"{*total_in_use_cnt*}"
	}

where:

Element	Description
`dynamic_add_cnt`	The number of dynamically learnt MAC addresses (integer).
`static_add_cnt`	The number of statically configured MAC addresses (integer).

Element	Description
`multicast_add_cnt`	The number of multicast MAC addresses (integer).
`total_in_use_cnt`	The total number of MAC addresses (integer).

Get Global FDB Runtime Settings

Gets the global FDB runtime settings.

Request

Method Type	GET
Request URI	/nos/api/info/fdb/global
Request Body (JSON)	

Response

Response Body (JSON)	{
	"global_learning_status":"{enabled|disabled}",
	"aging_time":"{*aging_time*}"
	}

where:

Element	Description
`global_learning_status`	The status of global MAC address learning; one of *enabled, disabled.*
`aging_time`	The MAC address aging time, in seconds; an integer from *0-1000000.*

Get Global FDB Configured Settings

Gets the global FDB configured settings.

Request

Method Type	GET
Request URI	/nos/api/cfg/fdb/global
Request Body (JSON)	

Response

Response Body (JSON)	{
	"global_learning_status":"{enabled|disabled}",
	"aging_time":"{*aging_time*}"
	}

where:

Element	Description
`global_learning_status`	The status of global MAC address learning; one of *enabled, disabled.*
`aging_time`	The MAC address aging time, in seconds; an integer from *0-1000000.*

Update Global FDB Settings

Updates the global FDB settings.

Request

Method Type	PUT
Request URI	/nos/api/cfg/fdb/global

Request Body (JSON)	{
	"global_learning_status":"{enabled|disabled}",
	"aging_time":"{*aging_time*}"
	}

where:

Element	Description
`global_learning_status`	The status of global MAC address learning; one of *enabled, disabled.*
`aging_time`	The MAC address aging time, in seconds; an integer from *0-1000000.*

Response

Response Body (JSON)	{
	"global_learning_status":"{enabled|disabled}",
	"aging_time":"{*aging_time*}"
	}

where:

Element	Description
`global_learning_status`	The status of global MAC address learning; one of *enabled, disabled.*
`aging_time`	The MAC address aging time, in seconds; an integer from *0-1000000.* **Note:** Setting this variable to 0 disables MAC address aging.

Get MAC Address Learning Interface Runtime Setting

Gets the runtime setting of MAC learning for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/fdb/interface/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Response

Response Body (JSON)	`{`
	`  "learning_status":"{enabled|disabled}"`
	`}`

where:

Element	Description
`learning_status`	The status of MAC address learning for the specified interface; one of *enabled, disabled.*

Get MAC Address Learning Interface Configured Setting

Gets the configured setting of MAC learning for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/fdb/interface/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Response

Response Body (JSON)	{
	"learning_status":"{enabled|disabled}"
	}

where:

Element	Description
`learning_status`	The status of MAC address learning for the specified interface; one of: *enabled, disabled.*

Update Interface MAC Address Learning Setting

Gets the status of MAC learning for a specific switch interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/fdb/interface/<*if_name*>

Request Body (JSON)	`{`
	`  "learning_status":"{enabled|disabled}"`
	`}`

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.
`learning_status`	The status of MAC address learning for the specified interface; one of *enabled, disabled.*

Response

Response Body (JSON)	`{`
	`  "learning_status":"{enabled|disabled}"`
	`}`

where:

Element	Description
`learning_status`	The status of MAC address learning for the specified interface; one of *enabled, disabled.*

Get Static MAC Addresses

Gets all statically configured MAC addresses.

Request

Method Type	GET
Request URI	/nos/api/cfg/fdb
Request Body (JSON)	

Response

Response Body (JSON)	{
	"address_table":
	[
	{
	"vlan_id":"{*vlan_id*}",
	"mac_address":"{*mac_address*}",
	"is_static":"{true|false}",
	"if_name":"{*if_name*}"
	}
	]
	}

where:

Element	Description
`address_table`	The MAC table entries.
`vlan_id`	The VLAN number for the MAC table entry; an integer from *1-4094.*
`mac_address`	The MAC address.
`is_static`	Whether the MAC address is statically configured; one of *true, false.*
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.

Create Static MAC Address

Adds a new statically configured MAC address.

Request

Method Type	POST
Request URI	/nos/api/cfg/fdb

Request Body (JSON)	{
	"mac_address":"{*mac_address*}",
	"vlan_id":"{*vlan_id*}",
	"interfaces":["{*if_name*}"]
	}

where:

Element	Description
mac_address	The MAC address.
vlan_id	The VLAN number; an integer from *1-4094*.
interfaces	The name of the switch interface. For example: *Ethernet1/12*.

Response

Response Body (JSON)	{
	"mac_address":"{*mac_address*}",
	"vlan_id":"{*vlan_id*}",
	"is_static":"{true|false}",
	"if_name":"{*if_name*}"
	}

where:

Element	Description
mac_address	The MAC address.
vlan_id	The VLAN number; an integer from *1-4094*.
is_static	Whether the MAC address is statically configured; one of *true, false*.
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Delete MAC Address or Interface for Multicast MAC Address

Removes a MAC address matching the search criteria, or removes a switch interface from a multicast MAC address.

Request

Method Type	PUT
Request URI	/nos/api/cfg/fdb

Request Body (JSON)	{
	"fdb_type":"{static|dynamic}",
	"mac_address":"{*mac_address*}",
	"interfaces":["{*if_name*}"],
	"vlan_id":"{*vlan_id*}"
	}

where:

Element	Description
`fdb_type`	The type of MAC address to delete; one of *static, dynamic.*
`mac_address`	The MAC address matching the criteria.
`interfaces`	The name of the switch interface to filter on. For example *Ethernet1/12*.
`vlan_id`	The VLAN number to filter on; an integer from *1-4094.*

HSC

Note: Hardware Switch Controller (HSC) is supported on the following Lenovo switches:

- RackSwitch G8272
- RackSwitch G8332
- RackSwitch G8296
- ThinkSystem NE1032T RackSwitch
- ThinkSystem NE1032 RackSwitch
- ThinkSystem NE1072T RackSwitch
- ThinkSystem NE10032 RackSwitch
- ThinkSystem NE2572 RackSwitch

The following HSC related URIs are available:

/nos/api/cfg/hsc/mode	POST, PUT, GET
/nos/api/cfg/hsc/devicename	POST, PUT, GET
/nos/api/cfg/hsc/ha	POST, PUT, GET
/nos/api/cfg/hsc/controller	POST, PUT, GET
/nos/api/cfg/hsc/tunnel	POST, PUT, GET
/nos/api/cfg/hsc/vtep	POST, PUT
/nos/api/cfg/hsc/vtep-port	POST, PUT
/nos/api/info/hsc/controller-connection	GET
/nos/api/info/hsc/restc-connection	GET
/nos/api/info/hsc/vtep	GET
/nos/api/info/hsc/mac-address	GET
/nos/api/info/hsc/tunnel	GET
/nos/api/info/hsc/virtual-net	GET
/nos/api/info/hsc/virtual-port	GET

The following HSC commands are available:

- Configure HSC Mode
- Update HSC Mode
- Configure Device Name
- Update Device Name
- Configure HSC HA Mode
- Update HSC HA Mode
- Configure HSC Controller
- Update HSC Controller

- Configure HSC Tunnel
- Update HSC Tunnel
- Configure HSC VTEP
- Update HSC VTEP
- Configure HSC VTEP Port
- Update HSC VTEP Port
- Get HSC Mode
- Get Device Name
- Get HSC HA Mode
- Get HSC Controller
- Get HSC Tunnel
- Get HSC Controller-Connection Information
- Get HSC RESTC-Connection Information
- Get HSC VTEP Basic Information
- Get HSC VTEP MAC-Address Information
- Get HSC VTEP Tunnel Information
- Get HSC VTEP Virtual-Net Information
- Get HSC VTEP Virtual-Port Information

Configure HSC Mode

Enables or disables HSC mode.

Request

Method Type	POST
Request URI	/nos/api/cfg/hsc/mode

Request Body
 	
(JSON)
	{
	"mode": "<mode>"
	}

where:

Element	Description
`mode`	Set HSC mode: • *vtep* sets the HSC mode to VTEP • *none* disables HSC mode

Response

Response Body
 	
(JSON)
	{
	"mode": "<mode>"
	}

where:

Element	Description
`mode`	Set HSC mode: • *vtep* sets the HSC mode to VTEP • *none* disables HSC mode

Update HSC Mode

Updates HSC mode.

Request

Method Type	PUT
Request URI	/nos/api/cfg/hsc/mode

Request Body
 	
(JSON)
	{
	"mode": "<mode>"
	}

where:

Element	Description
`mode`	Set HSC mode: • *vtep* sets the HSC mode to VTEP • *none* disables HSC mode

Response

Response Body
 	
(JSON)
	{
	"mode": "<mode>"
	}

where:

Element	Description
`mode`	Set HSC mode: • *vtep* sets the HSC mode to VTEP • *none* disables HSC mode

Configure Device Name

Configures the HSC device name.

Request

Method Type	POST
Request URI	/nos/api/cfg/hsc/devicename

Request Body
 	
(JSON)
	{
	“device-name”: “<device-name>”
	}

where:

Element	Description
device-name	Set the device name using <device-name> (string). A null string (“”) deletes the device name.

Response

Response Body
 	
(JSON)
	{
	"device-name": "<device-name>"
	}

where:

Element	Description
device-name	Set the device name using <device-name> (string). A null string (“”) deletes the device name.

Update Device Name

Updates the HSC device name.

Request

Method Type	PUT
Request URI	/nos/api/cfg/hsc/devicename

Request Body
 	
(JSON)
	{
	“device-name”: “<device-name>”
	}

where:

Element	Description
device-name	Set the device name using <device-name> (string). A null string (“”) deletes the device name.

Response

Response Body
 	
(JSON)
	{
	"device-name": "<device-name>"
	}

where:

Element	Description
device-name	Set the device name using <device-name> (string). A null string (“”) deletes the device name.

Configure HSC HA Mode

Configures the HSC HA mode.

Request

Method Type	POST
Request URI	/nos/api/cfg/hsc/ha

Request Body
 	
(JSON)
	{
	"ha-mode": "<ha-mode>"
	}

where:

Element	Description
`ha-mode`	Set the HA mode: • *vlag* sets the HSC mode to VLAG • *none* disables HA mode

Response

Response Body
 	
(JSON)
	{
	"ha-mode": "<ha-mode>"
	}

where:

Element	Description
`ha-mode`	Set the HA mode: • *vlag* sets the HSC mode to VLAG • *none* disables HA mode

Update HSC HA Mode

Updates the HSC HA mode.

Request

Method Type	PUT
Request URI	/nos/api/cfg/hsc/ha

Request Body
 	
(JSON)
	{
	"ha-mode": "<ha-mode>"
	}

where:

Element	Description
`ha-mode`	Set the HA mode: • *vlag* sets the HSC mode to VLAG • *none* disables HA mode

Response

Response Body
 	
(JSON)
	{
	"ha-mode": "<ha-mode>"
	}

where:

Element	Description
`ha-mode`	Set the HA mode: • *vlag* sets the HSC mode to VLAG • *none* disables HA mode

Configure HSC Controller

Configures the HSC controller.

Request

Method Type	POST
Request URI	/nos/api/cfg/hsc/controller

Request Body
 	
(JSON)
	{
	“provider” : “nsx”,
	“IP” : “A.B.C.D”,
	“port” : 6640,
	“vrf” : “management”
	}

where:

Element	Description
`provider`	A specified string value which sets the controller provider: • *nsx* sets the provider to VMware NSX Controller • *sdnc* sets the provider to Lenovo SDN Controller • *none* remove all the controller configuration
`IP`	The Controller IP address.
`port`	The port number; an integer from *1-65535*. Default value: *6640*.
`vrf`	A specified string value which determines the VRF used to connect to the controller; one of: • *management* to use management VRF • *default* to use default VRF Default value: *management*.

Response

Response Body
 	
(JSON)
	{
	“provider” : “nsx”,
	“IP” : “A.B.C.D”,
	“port” : 6640,
	“vrf” : “management”
	}

where:

Element	Description
`provider`	A specified string value which sets the controller provider: • *nsx* sets the provider to VMware NSX Controller • *sdnc* sets the provider to Lenovo SDN Controller • *none* remove all the controller configuration
`IP`	The Controller IP address.
`port`	The port number; an integer from *1-65535*. Default value: *6640*.
`vrf`	A specified string value which determines the VRF used to connect to the controller; one of: • *management* to use management VRF • *default* to use default VRF Default value: *management*.

Update HSC Controller

Updates the HSC controller.

Request

Method Type	PUT
Request URI	/nos/api/cfg/hsc/controller

Request Body
 	
(JSON)
	{
	“provider” : “nsx”,
	“IP” : “A.B.C.D”,
	“port” : 6640,
	“vrf” : “management”
	}

where:

Element	Description
`provider`	A specified string value which sets the controller provider: • *nsx* sets the provider to VMware NSX Controller • *sdnc* sets the provider to Lenovo SDN Controller • *none* remove all the controller configuration
`IP`	The Controller IP address.
`port`	The port number; an integer from *1-65535*. Default value: *6640*.
`vrf`	A specified string value which determines the VRF used to connect to the controller; one of: • *management* to use management VRF • *default* to use default VRF Default value: *management*.

Response

Response Body
 	
(JSON)
	{
	“provider” : “nsx”,
	“IP” : “A.B.C.D”,
	“port” : 6640,
	“vrf” : “management”
	}

where:

Element	Description
`provider`	A specified string value which sets the controller provider: • *nsx* sets the provider to VMware NSX Controller • *sdnc* sets the provider to Lenovo SDN Controller • *none* remove all the controller configuration
`IP`	The Controller IP address.
`port`	The port number; an integer from *1-65535*. Default value: *6640*.
`vrf`	A specified string value which determines the VRF used to connect to the controller; one of: • *management* to use management VRF • *default* to use default VRF Default value: *management*.

Configure HSC Tunnel

Configures the HSC tunnel.

Request

Method Type	POST
Request URI	/nos/api/cfg/hsc/tunnel

Request Body
(JSON)
	{
	“tunnel-IP”: “A.B.C.D”
	}

where:

Element	Description
`tunnel-IP`	The IP address to be used as tunnel IP. "*A.B.C.D*" sets the tunnel IP to a valid IPv4 address. A null string (“”) deletes the configured tunnel IP.

Response

Response Body
(JSON)
	{
	“tunnel-IP”: “A.B.C.D”
	}

where:

Element	Description
`tunnel-IP`	The IP address to be used as tunnel IP. "*A.B.C.D*" sets the tunnel IP to a valid IPv4 address. A null string (“”) deletes the configured tunnel IP.

Update HSC Tunnel

Updates the HSC tunnel.

Request

Method Type	PUT
Request URI	/nos/api/cfg/hsc/tunnel

Request Body
 	
(JSON)
	{
	  “tunnel-IP”: “A.B.C.D”
	}

where:

Element	Description
`tunnel-IP`	The IP address to be used as tunnel IP. "*A.B.C.D*" sets the tunnel IP to a valid IPv4 address. A null string (“”) deletes the configured tunnel IP.

Response

Response Body
 	
(JSON)
	{
	  “tunnel-IP”: “A.B.C.D”
	}

where:

Element	Description
`tunnel-IP`	The IP address to be used as tunnel IP. "*A.B.C.D*" sets the tunnel IP to a valid IPv4 address. A null string (“”) deletes the configured tunnel IP.

Configure HSC VTEP

Configures the HSC VTEP.

Request

Method Type	POST
Request URI	/nos/api/cfg/hsc/vtep

Request Body
 	
(JSON)
	{
	“vtep-id” : <1-2>,
	“IP” : “a.b.c.d”,
	“port”: <1-65535>,
	“vrf” : “<management|default>”,
	“username” : “<string>”,
	“password” : “<string>”
	}

where:

Element	Description
`vtep-id`	The VTEP ID; one of *1,2*.
`IP`	The local VTEP IP address used to connect with HSC. Entering no IP address deletes the specified tunnel IP.
`port`	The port number; an integer from *1-65535*. Default value: *443*.
`vrf`	A specified string value which determines the VRF used for the HSC - local VTEP connection; one of: • *management* to use management VRF • *default* to use default VRF Default value: *management*.
`username`	The username. Default value: *admin*.
`password`	The password. Default value: *admin*.

Response

`True` if the operation succeeded; otherwise `False`.

Update HSC VTEP

Updates the HSC VTEP.

Request

Method Type	PUT
Request URI	/nos/api/cfg/hsc/vtep

Request Body
 	
(JSON)
	{
	“vtep-id” : <1-2>,
	“IP” : “a.b.c.d”,
	“port”: <1-65535>,
	“vrf” : “<management|default>”,
	“username” : “<string>”,
	“password” : “<string>”
	}

where:

Element	Description
`vtep-id`	The VTEP ID; one of *1,2*.
`IP`	The local VTEP IP address used to connect with HSC. Entering no IP address deletes the specified tunnel IP.
`port`	The port number; an integer from *1-65535*. Default value: *443*.
`vrf`	A specified string value which determines the VRF used for the HSC - local VTEP connection; one of: • *management* to use management VRF • *default* to use default VRF Default value: *management*.
`username`	The username. Default value: *admin*.
`password`	The password. Default value: *admin*.

Response

`True` if the operation succeeded; otherwise `False`.

Configure HSC VTEP Port

Configures the HSC VTEP port.

Request

Method Type	POST
Request URI	/nos/api/cfg/hsc/vtep-port

Request Body
 	
(JSON)
	{
	“vtep-id” : <1-2>,
	“vxlan-port”:
	{
	“eth”:{“action”: “add”, port-list : “1/3, 1/5-10”},
	“aggregation”: {“action”: “add”, “port-list”: “2,4,5-8”},
	“vlag-instance”: {“action”: “add”, “port-list”: “1,5,7-9”}
	}
	}

where:

Element	Description
`vtep-id`	The VTEP ID; one of *1,2*.
`action`	The configuration type: • *add* to add a new port. • *remove* to remove a port from the current configuration. • *set* to set the current configuration with port-list; this over-writes the current VXLAN port.
`vxlan-port`	Configure the VXLAN enabled interfaces; one of: • *eth* enables VXLAN on an ethernet interface • *aggregation* enables VXLAN on an aggregation interface • *vlag-instance* enables VXLAN on a VLAG instance

Response

`True` if the operation succeeded; otherwise `False`.

Update HSC VTEP Port

Updates the HSC VTEP port.

Request

Method Type	PUT
Request URI	/nos/api/cfg/hsc/vtep-port

Request Body
 	
(JSON)
	{
	“vtep-id” : <1-2>,
	“vxlan-port”:
	{
	“eth”:{“action”: “add”, port-list : “1/3, 1/5-10”},
	“aggregation”: {“action”: “add”, “port-list”: “2,4,5-8”},
	“vlag-instance”: {“action”: “add”, “port-list”: “1,5,7-9”}
	}
	}

where:

Element	Description
`vtep-id`	The VTEP ID; one of *1,2*.
`action`	The configuration type: • *add* to add a new port. • *remove* to remove a port from the current configuration. • *set* to set the current configuration with port-list; this over-writes the current VXLAN port.
`vxlan-port`	Configure the VXLAN enabled interfaces; one of: • *eth* enables VXLAN on an ethernet interface • *aggregation* enables VXLAN on an aggregation interface • *vlag-instance* enables VXLAN on a VLAG instance

Response

`True` if the operation succeeded; otherwise `False`.

Get HSC Mode

Returns the HSC mode.

Request

Method Type	GET
Request URI	/nos/api/cfg/hsc/mode
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	    "mode": "<mode>"
	}</pre>

where:

Element	Description
`mode`	Set HSC mode: • *vtep* sets the HSC mode to VTEP • *none* disables HSC mode

Get Device Name

Returns the HSC device name.

Request

Method Type	GET
Request URI	/nos/api/cfg/hsc/devicename
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"device-name": "<device-name>"
	}

where:

Element	Description
`device-name`	Set the device name using `<device-name>` (string). A null string (“”) deletes the device name.

Get HSC HA Mode

Returns the HSC HA mode.

Request

Method Type	GET
Request URI	/nos/api/cfg/hsc/ha
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	“ha-mode”: “vlag”
	}

where:

Element	Description
`ha-mode`	Set the HA mode: • *vlag* sets the HSC mode to VLAG • *none* disables HA mode

Get HSC Controller

Returns the HSC controller.

Request

Method Type	GET
Request URI	/nos/api/cfg/hsc/controller
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	“provider” : “nsx”,
	“IP” : “A.B.C.D”,
	“port” : 6640,
	“vrf” : “management”
	}

where:

Element	Description
`provider`	A specified string values which sets the controller provider: • *nsx* sets the provider to VMware NSX Controller • *sdnc* sets the provider to Lenovo SDN Controller • *none* remove all the controller configuration
`IP`	The Controller IP address.
`port`	The port number; an integer from *1-65535*. Default value: *6640*.
`vrf`	A specified string value which determines the VRF used to connect to the controller; one of: • *management* to use management VRF • *default* to use default VRF Default value: *management*.

Get HSC Tunnel

Returns the HSC controller.

Request

Method Type	GET
Request URI	/nos/api/cfg/hsc/tunnel
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  “tunnel-IP”: “A.B.C.D”
	}

where:

Element	Description
`tunnel-IP`	The IP address to be used as tunnel IP. "*A.B.C.D*" sets the tunnel IP to a valid IPv4 address. A null string (“”) deletes the configured tunnel IP.

Get HSC Controller-Connection Information

Returns the HSC OVSDB-connection information.

Request

Method Type	GET
Request URI	/nos/api/info/hsc/controller-connection
Request Body (JSON)	

Response

Response Body (JSON)	[{"type":"SSL (NSX Controller)", "peer":"<a.b.c.d:port>", "inact":<int>, "backoff":<int>}, {"type":"SSL (NSX Controller)", "peer":"<a.b.c.d:port>", "inact":<int>, "backoff":<int>}, …]

where:

Element	Description
`type`	String value for Connection Type. For example: *SSL (NSX Controller.*
`inact`	The inactive probe time in milliseconds. If the value displayed is `-1`, the inactive probe time is invalid.
`backoff`	The maximum backoff time in milliseconds. `-1` means invalid.

Get HSC RESTC-Connection Information

Returns the API RESTC-connection information.

Request

Method Type	GET
Request URI	/nos/api/info/hsc/restc-connection
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	`[`
	` {"owner":"vtep1", "peer":"<{https|http}://a.b.c.d:port>",`
	`"vrf":"<vrf-name>", "state":"ready|unready"},`
	` {"owner":"vtep2", "peer":"<{https|http}://a.b.c.d:port>",`
	`"vrf":"<vrf-name>", "state":"ready|unready"}`
	`]`

where:

Element	Description
`owner`	The owner; one of *vtep1, vtep2*.
`peer`	The peer address in the following format: *{https|http}://a.b.c.d:port*.
`vrf`	A specified string value which determines the VRF used for the HSC - local VTEP connection; one of: • *management* to use management VRF • *default* to use default VRF
`state`	The state; one of: • *init* HSCD is in initiate state • *logging* connecting to NWVD, HSDC rest client is in logging state • *running* connection to NWVD has been created, HSDC rest client is running state • *unready* NWVD is not ready, HSDC is in checking state

Get HSC VTEP Basic Information

Returns the HSC VTEP information.

Request

Method Type	GET
Request URI	/nos/api/info/hsc/vtep
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	    "status" : "Enabled",
	    "ha-mode": "vlag",
	    "device-name" : "<name>",
	    "tunnel-ip": "a.b.c.d",
	    "bfd-status": "Enabled",
	    "physical-port-count":<int>,
	    "total-mapping-count": <int>
	}</pre>

where:

Element	Description
`status`	The status; one of *enabled, disabled*.
`ha-mode`	The HA- mode; one of *vlag, none*.
`device-name`	The HSC device name.
`tunnel-IP`	The local tunnel IP address for NSXGW.
`bfd-status`	The BFD status value; one of *enabled, disabled*.
`physical-port`	The physical port count.
`total-mapping-count`	The total mappings count.

Get HSC VTEP MAC-Address Information

Returns the HSC VTEP MAC-address information.

Request

Method Type	GET
Request URI	/nos/api/info/hsc/mac-address
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	    "local":
	    {
	        "count":<count>,
	        "mac-table":
	        [
	            {"vni":<vni>, "mac":"<mac-address>",
	"tunnel":"a.b.c.d"},
	            {"vni":<vni>, "mac":"<mac-address>",
	"tunnel":"a.b.c.d"},
	            ...
	        ]
	    },
	
	    "remote":
	    {
	        "count":<count>,
	        "mac-table":
	        [
	            {"vni":<vni>, "mac":"<mac-address>",
	"tunnel":"a.b.c.d"},
	            {"vni":<vni>, "mac":"<mac-address>",
	"tunnel":"a.b.c.d"},
	            ...
	        ]
	    },
	
	    "local-mcast":
	    {
	        "count":<count>,
	        "mac-table":
	        [
	            {"vni":<vni>, "mac":"<mac-address>",
	"tunnel":"a.b.c.d"},
	            {"vni":<vni>, "mac":"<mac-address>",
	"tunnel":"a.b.c.d"},
	            ...
	        ]
	    },
	
	    "remote-mcast":
	    {
	        "count":<count>,
	        "mac-table":
	        [
	            {"vni":<vni>, "mac":"<mac-address>",
	"tunnel":"a.b.c.d"},
	            {"vni":<vni>, "mac":"<mac-address>",
	"tunnel":"a.b.c.d"},
	            ...
	        ]
	    }
	}</pre>

where:

Element	Description
`local`	The unicast MAC address table information of local sites.
`remote`	The unicast MAC address table information of remote sites.
`local-mcast`	The multicast MAC table information of local sites.
`remote-mcast`	The multicast MAC table information of remote sites.
`count`	The number of the listed MAC-table items.
`vni`	The VXLAN Network Identifier; an integer from *1-16777214*.
`mac`	The MAC address value string.
`tunnel`	The tunnel IP address string.

Get HSC VTEP Tunnel Information

Returns the VTEP tunnel information.

Request

Method Type	GET
Request URI	/nos/api/info/hsc/tunnel
Request Body (JSON)	

Response

Response Body (JSON)	[{"local-ip":"a.b.c.d, "remote-ip":"a.b.c.d", "rmt-bfd-enabled":"<true/false>"}, {"local-ip":"a.b.c.d, "remote-ip":"a.b.c.d", "rmt-bfd-enabled":"<true/false>"}, ...]

where:

Element	Description
`local-ip`	IP address of local switch. This can be the management IP address of the local switch.
`remote-ip`	The remote IP address of the tunnel.
`rmt-bfd-enabled`	The remote BFD status; one of *true, false*.

Get HSC VTEP Virtual-Net Information

Returns the VTEP virtual-net information.

Request

Method Type	GET
Request URI	/nos/api/info/hsc/virtual-net
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	`[`
	`  {"vni":<vni>, "name":"<name>"},`
	`  {"vni":<vni>, "name":"<name>"},`
	`  ...`
	`]`

where:

Element	Description
`vni`	The VXLAN Network Identifier; an integer from *1-16777214*.
`name`	The unique string value name of the virtual network.

Get HSC VTEP Virtual-Port Information

Returns the VTEP virtual-port information.

Request

Method Type	GET
Request URI	/nos/api/info/hsc/virtual-port
Request Body (JSON)	

Response

Response Body (JSON)	[{"name":"<port name>", "vlan": <vlanid>, "vnid":<vnid>}, {"name":"<port name>", "vlan": <vlanid>, "vnid":<vnid>}, ...]

where:

Element	Description
`name`	The virtual port name.
`vlan`	The Vlan ID.
`vnid`	The VXLAN Network Identifier; an integer from *1-16777214*.

IGMP

The following Internet Group Management Protocol (IGMP) URIs are available:

- /nos/api/cfg/mc_vlan/groups?vid="<*vid*>"&if_name="<*if_name*>" GET
- /nos/api/cfg/mc_vlan/mrouter?vid="<*vid*>"&if_name="<*if_name*>" GET
- /nos/api/cfg/mc_vlan/mrouter/<*vlan_id*> PUT
- /nos/api/cfg/mc_vlan/querier/<*vlan_id*> GET, PUT
- /nos/api/cfg/igmp/snoop GET, PUT
- /nos/api/cfg/mc_vlan GET
- /nos/api/cfg/mc_vlan/<*vlan_id*> GET, PUT

The following IGMP commands are available:

- Get IGMP Groups
- Get IGMP Mrouter
- Update IGMP Mrouter Interface for a VLAN
- Get IGMP Querier
- Update IGMP Querier on a VLAN
- Get IGMP Snooping System Properties
- Update IGMP Snooping System Properties
- Get IGMP Snooping Properties of All VLANs
- Get IGMP Snooping VLAN Properties
- Update IGMP Snooping VLAN Properties

Get IGMP Groups

Gets all Internet Group Management Protocol (IGMP) snooping groups' membership information for a specific VLAN or interface/port aggregation.

Notes:

If the specified *vid* is `None`, this request gets a list of IGMP snooping groups' membership information for all VLAN.

If the specified *if_name* is `None`, this request gets a list of IGMP snooping groups' membership information for all interface/port aggregations.

A value must be provided for either vid or if_name in the request.

Request

Method Type	GET
Request URI	/nos/api/cfg/mc_vlan/groups?vid="<*vid*>"&if_name="<*if_name*>"
Request Body (JSON)	

where:

Element	Description
vid	VLAN number; an integer from 1-3999.
if_name	Ethernet interface name or port aggregation name.

Response

Response Body

(JSON)

```
[
  {
    "vid": "<vid>",
    "if_name": "<if_name>",
    "group_address": "<group_address>",
    "source_ip":
    {
      "include_list" :
      [
        {
          "source_address" : "<source ip address>",
          "uptime" : "<uptime>",
          "expires" : "<expires>",
          "fwd" : "<fwd>",
          "flags" : "<flags>"
        }
      ],
      "exclude_list" :
      [
        {
          "source_address" : "<source ip address>",
          "uptime" : "<uptime>",
          "expires" : "<expires>",
          "fwd" : "<fwd>",
          "flags" : "<flags>"
        }
      ]
    }
    "flags" : "<flags>",
    "expires": "<expires>",
    "version": "<version>",
    "filter_mode": "<filter_mode>",
  }
]
```

where:

Element	Description
`vid`	VLAN number; an integer from 1-3999.
`if_name`	Ethernet interface name or port aggregation name.
`group_address`	The IGMP group IPv4 address.
`source_ip`	Dictionary of included and excluded source IP details.
`include_list`	List of included source IP details.
`exclude_list`	List of excluded source IP details.
`source_address`	Included or excluded source IPv4 address.
`uptime`	Time since switch is running in the following format: HH:MM:SS.
`expires`	Source expiry time interval in the following format: HH:MM:SS.

Element	Description
`fwd`	Whether to forward traffic for this source IP; one of *yes, no*.
`flags`	Source IP flag; one of: • `D` – Dynamic • `S` – Static • `L` – Local and Static **Note:** This flag is included in `include_list` and `exclude_list`.
`flags`	Group flags; one of *Dynamic, Static IGMP group*.
`expires`	Group expiry time interval in the following format: HH:MM:SS.
`version`	IGMP version number.
`filter_mode`	IGMP Router-Filter-Mode State; one of *include, exclude*.

Get IGMP Mrouter

Gets Internet Group Management Protocol (IGMP) Multicast Router (mrouter) entries for a specific VLAN or interface/port-aggregation.

Request

Method Type	GET
Request URI	/nos/api/cfg/mc_vlan/mrouter?vid="<*vlan_id*>"&if_name="<*if_name*>"
Request Body (JSON)	

where:

Element	Description
vlan_id	VLAN number; an integer from 1-3999.
if_name	Ethernet interface name or port aggregation name.

Response

Response Body
 	
(JSON)
	[
	  {
	  "vid": "<*vid*>",
	  "if_name": "<*if_name*>",
	  "mrouter_address": "<*mrouter_address*>",
	  "mrouter_type": "<*mrouter_type*>",
	  "expires": "<*expires*>",
	  }
	]

where:

Element	Description
`vid`	VLAN number; an integer from *1-3999*.
`if_name`	Ethernet interface name or port aggregation name.
`mrouter_address`	The IGMP multicast router IPv4 address.
`mrouter_type`	Specifies how this entry was learned; one of *dynamic, PIM hello, static*.
`expires`	Expiry time interval in the following format: HH:MM:SS.

Update IGMP Mrouter Interface for a VLAN

Adds a Layer 2 interface as a static multicast router port.

Request

Method Type	PUT
Request URI	/nos/api/cfg/mc_vlan/mrouter/<*vid*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	}

where:

Element	Description
vid	VLAN number; an integer from 1-3999.
if_name	Ethernet interface name or port aggregation name.

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	}

where:

Element	Description
`if_name`	Ethernet interface name or port aggregation name.

Get IGMP Querier

Gets IGMP querier information for a VLAN or for all VLANs.

Request

Method Type	GET
Request URI	/nos/api/cfg/mc_vlan/querier/<*vlan_id*>"
Request Body (JSON)	

where:

Element	Description
vlan_id	VLAN number; an integer from 1-3999.

Response

Response Body
 	
(JSON)
	[
	  {
	  "vid": "<*vid*>",
	  "address": "<*address*>",
	  "state": "<*state*>",
	  "version": "<*version*>",
	  "expires": "<*expires*>",
	  }
	]

where:

Element	Description
`vid`	VLAN number; an integer from *1-3999*.
`address`	Querier IPv4 address.
`state`	Elected querier state; one of *Querier, Non-querier*.
`version`	Snooping querier version.
`expires`	Expiry time interval in the following format: HH:MM:SS.

Update IGMP Querier on a VLAN

Enables or disables the Internet Group Management Protocol (IGMP) snooping querier on the specified VLAN.

Note: To set the querier address for a VLAN, the VLAN need not be present. After the VLAN is created, the running configuration will show the querier address update.

Request

Method Type	PUT
Request URI	/nos/api/cfg/mc_vlan/querier/<*vid*>

Request Body
 	
(JSON)
	{
	"address": "<*address*>",
	}

where:

Element	Description
`address`	The querier IPv4 address.

Response

Response Body
 	
(JSON)
	[
	{
	"vid": "<*vid*>",
	"address": "<*address*>",
	"state": "<*state*>",
	"version": "<*version*>",
	"expires": "<*expires*>",
	}
	]

where:

Element	Description
`vid`	VLAN number; an integer from *1-3999*.
`address`	Querier IPv4 address.
`state`	Elected querier state; one of *Querier, Non-querier*.
`version`	Snooping querier version.
`expires`	Expiry time interval in the following format: HH:MM:SS.

Get IGMP Snooping System Properties

Gets global IGMP Snooping properties of the system.

Request

Method Type	GET
Request URI	/nos/api/cfg/igmp/snoop
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"ena_igmp_snoop": "*<ena_igmp_snoop>*"
	}

where:

Element	Description
`ena_igmp_snoop`	Enables IGMP snooping globally on all VLANs; one of *yes, no*. Default value: *yes.* If disabled globally, IGMP snooping is disabled on all VLANs, regardless of the per-VLAN setting of IGMP snooping. If IGMP snooping is enabled globally, the per-VLAN setting of IGMP snooping takes effect.

Update IGMP Snooping System Properties

Updates the global IGMP Snooping properties of the system.

Request

Method Type	PUT
Request URI	/nos/api/cfg/igmp/snoop

Request Body
 	
(JSON)
	{
	"ena_igmp_snoop": "*<ena_igmp_snoop>*"
	}

where:

Element	Description
`ena_igmp_snoop`	Enables IGMP snooping globally on all VLANs; one of *yes, no*. Default value: *yes.* If disabled globally, IGMP snooping is disabled on all VLANs, regardless of the per-VLAN setting of IGMP snooping. If IGMP snooping is enabled globally, the per-VLAN setting of IGMP snooping takes effect.

ResponseResponse

Response Body
 	
(JSON)
	{
	"ena_igmp_snoop": "*<ena_igmp_snoop>*"
	}

Get IGMP Snooping Properties of All VLANs

Gets the IGMP snooping properties of all VLANs.

Request

Method Type	**GET**
Request URI	/nos/api/cfg/mc_vlan
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "vlan_id": "`*<vlan_id>*`",`
	`  "ena_igmp_snoop": "`*<ena_igmp_snoop>*`"`
	`  }`
	`]`

where:

Element	**Description**
`vlan_id`	VLAN number.
`ena_igmp_snoop`	Enables IGMP snooping on a VLAN; one of *yes, no*. Default value: *yes.*

Get IGMP Snooping VLAN Properties

Gets the IGMP snooping properties of one VLAN.

Request

Method Type	GET
Request URI	/nos/api/cfg/mc_vlan/<*vlan_id*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "vlan_id": "<*vlan_id*>",
	  "ena_igmp_snoop": "<*ena_igmp_snoop*>"
	  }
	]

where:

Element	Description
`vlan_id`	VLAN number.
`ena_igmp_snoop`	Enables IGMP snooping on a VLAN; one of *yes, no*. Default value: *yes*.

Update IGMP Snooping VLAN Properties

Updates the IGMP snooping properties of the specified VLAN.

Request

Method Type	PUT
Request URI	/nos/api/cfg/mc_vlan/<*vlan_id*>

Request Body
 	
(JSON)
	{
	"vlan_id": "<*vlan_id*>",
	"ena_igmp_snoop": "<*ena_igmp_snoop*>",
	"fast_leave": "<*fast_leave*>",
	"query_interval": "<*query_interval*>",
	"version": "<*version*>",
	}

where:

Element	Description
`vlan_id`	VLAN number. **Note:** The VLAN must exist.
`ena_igmp_snoop`	(Optional) Whether to enable IGMP snooping on a VLAN; one of *yes, no*. Default value: *yes.*
`fast_leave`	One of *yes, no*. Default value: *no.*
`query_interval`	(Optional) IGMP query interval, in seconds; an integer from *1-18000*. Default value: *125*.
`version`	(Optional) IGMP Snooping version number; one of *2, 3*. Default value: *3*.

Response

Response Body
 	
(JSON)
	{
	"vlan_id": "<*vlan_id*>",
	"ena_igmp_snoop": "<*ena_igmp_snoop*>",
	"fast_leave": "<*fast_leave*>",
	"query_interval": "<*query_interval*>",
	"version": "<*version*>",
	}

where:

Element	Description
`vlan_id`	VLAN number. **Note:** The VLAN must exist.
`ena_igmp_snoop`	(Optional) Whether to enable IGMP snooping on a VLAN; one of *yes, no*. Default value: *yes.*

Element	Description
`fast_leave`	One of *yes, no*. Default value: *no.*
`query_ interval`	(Optional) IGMP query interval, in seconds; an integer from *1-18000*. Default value: *125*.
`version`	(Optional) IGMP Snooping version number; one of *2, 3.* Default value: *3*.

Interface

The following interface URIs are available:

- /nos/api/cfg/interface GET
- /nos/api/cfg/interface/<*if_name*> GET, PUT
- /nos/api/cfg/interface/transceiver GET
- /nos/api/cfg/interface/transceiver/<*if_name*> GET

The following interface commands are available:

- Get All Interfaces
- Get Interface
- Update Interface
- Get Transceiver Information for All Interfaces
- Get Transceiver Information for One Interfaces

Get All Interfaces

Gets properties of all interfaces.

Request

Method Type Type	Get
Request URI	/nos/api/cfg/interface
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>[
	{
	  "duplex": "full",
	  "if_name": "Ethernet1/1",
	  "mtu": 1500,
	  "admin_state": "down",
	  "mac_addr": "a897.dc1b.8602",
	  "ifindex": 410001,
	  "oper_state": "down",
	  "speed": "40000"
	}
	]</pre>

where:

Element	Description
`if_name`	The interface name; a string. **Note:** The interface must exist.
`duplex`	The communication method of the interface; one of *auto, full, half*.
`speed`	The communication speed of the interface; one of the following: • *auto (auto negotiate)* • *10 (10Mb/s)* • *100 (100Mb/s)* • *1000 (1Gb/s)* • *10000 (10Gb/s)* • *25000 (10Gb/s)* • *40000 (40Gb/s).*
`mtu`	The maximum transmission unit, in bytes; a positive integer from *64-9216*.
`mac_addr`	The MAC address in the following format: xxxx.xxxx.xxxx.

Element	Description
admin_state	The admin status; one of *up, down*.
oper_state	The operation state; one of *up, down*.

Get Interface

Gets properties of one interface.

Request

Method Type Type	GET
Request URI	/nos/api/cfg/interface/*<if_name>*
Request Body (JSON)	

where:

Element	Description
if_name	The interface name; a string. **Note:** The interface must exist.

Response

Response Body
 	
(JSON)
	<pre>{
	  "duplex": "full",
	  "if_name": "Ethernet1/1",
	  "mtu": 1500,
	  "admin_state": "down",
	  "mac_addr": "a897.dc1b.8602",
	  "ifindex": 410001,
	  "oper_state": "down",
	  "speed": "40000"
	}</pre>

where:

Element	Description
`if_name`	The interface name; a string. **Note:** The interface must exist.
`duplex`	The communication method of the interface; one of *auto, full, half*.

Element	Description
`speed`	The communication speed of the interface; one of the following: • *auto (auto negotiate)* • *10 (10Mb/s)* • *100 (100Mb/s)* • *1000 (1Gb/s)* • *10000 (10Gb/s)* • *25000 (10Gb/s)* • *40000 (40Gb/s).*
`mtu`	The maximum transmission unit, in bytes; a positive integer from *64-9216*.
`mac_addr`	The MAC address in the following format: xxxx.xxxx.xxxx.
`admin_state`	The admin status; one of *up, down*.
`oper_state`	The operation state; one of *up, down*.

Update Interface

Updates properties of one interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/interface/<*if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"mtu": "<*mtu*>",
	"admin_state": "<*admin_state*>"
	}

where

Element	Description
if_name	The interface name (String). **Note:** The interface must exist.
mtu	The maximum transmission unit, in bytes; a positive integer from 64-9216. Default value: 1500.
admin_state	The admin status; up (default), down.

Note: If an element is not specified in a PUT request, no update for that element will be performed.

Response

Response Body
 	
(JSON)
	{
	"duplex": "full",
	"if_name": "Ethernet1/1",
	"mtu": 1500,
	"admin_state": "down",
	"mac_addr": "a897.dc1b.8602",
	"ifindex": 410001,
	"oper_state": "down",
	"speed": "40000"
	}

where:

Element	Description
`if_name`	The interface name; a string. **Note:** The interface must exist.
`duplex`	The communication method of the interface; one of *auto, full, half*.

Element	Description
`speed`	The communication speed of the interface; one of the following: • *auto (auto negotiate)* • *10 (10Mb/s)* • *100 (100Mb/s)* • *1000 (1Gb/s)* • *10000 (10Gb/s)* • *25000 (10Gb/s)* • *40000 (40Gb/s)*.
`if_name`	The interface name; a string. **Note:** The interface must exist.
`duplex`	The communication method of the interface; one of *auto, full, half*.
`speed`	The communication speed of the interface; one of the following: • *auto (auto negotiate)* • *10 (10Mb/s)* • *100 (100Mb/s)* • *1000 (1Gb/s)* • *10000 (10Gb/s)* • *40000 (40Gb/s)*.

Example

Method Type	PUT
Request URI	/nos/api/cfg/interface/Ethernet1%2F5

Request Body
 	
(JSON)
	{
	"if_name": "Ethernet1/5",
	"duplex": "duplex-full",
	"speed": "auto",
	"mtu": 9216,
	"mac_addr": "0001.0200.0005",
	"admin_state": "up"
	}

Get Transceiver Information for All Interfaces

Gets transceiver information for all interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/interface/transceiver
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>[
	 {
	  "status": "Disabled",
	  "part number": "00D5803-N13692A ",
	  "vendor": "IBM-Amphenol ",
	  "temperature": " 0.0C",
	  "volts": "0.00V",
	  "rev": "N/A",
	  "if_name": "Ethernet1/1",
	  "installed": "Present",
	  "serial number": "YK10FY382776 ",
	  "link": "Down",
	  "approval": "Approved",
	  "type": "40Gb Passive DAC 3m"
	 },
	]</pre>

where:

Element	Description
`if_name`	The IP interface name; a string. **Note:** The interface must exist.
`installed`	Whether the transceiver is present; one of *present, not present*.
`status`	Interface status; one of *enabled, disabled*.
`type`	Type of transceiver; a string.
`vendor`	Vendor of transceiver; a string.
`part_number`	Part number of transceiver; a string.
`revision`	Revision of transceiver; a string or N/A.
`serial number`	Serial number of transceiver; a string.
`volts`	Volts of transceiver; a string or N/A.

Element	Description
`temperature`	Temperature of transceiver; a string or N/A.
`approved`	Approval status of the transceiver; one of *approved, accepted, unapproved, Unsuport, Restrict, No Device*.

Get Transceiver Information for One Interfaces

Gets the transceiver information for a specified interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/interface/transceiver/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The IP interface name; a string. **Note:** The interface must exist.

Response

Response Body
 	
(JSON)
	[
	{
	"status": "Disabled",
	"part number": "00D5803-N13692A ",
	"vendor": "IBM-Amphenol ",
	"temperature": " 0.0C",
	"volts": "0.00V",
	"rev": "N/A",
	"if_name": "Ethernet1/1",
	"installed": "Present",
	"serial number": "YK10FY382776 ",
	"link": "Down",
	"approval": "Approved",
	"type": "40Gb Passive DAC 3m"
	}
	]

where:

Element	Description
`if_name`	The IP interface name; a string. **Note:** The interface must exist.
`installed`	Whether the transceiver is present; one of *present, not present*.
`status`	Interface status; one of *enabled, disabled*.
`type`	Type of transceiver; a string.
`vendor`	Vendor of transceiver; a string.

Element	Description
part_number	Part number of transceiver; a string.
revision	Revision of transceiver; a string or N/A.
serial number	Serial number of transceiver; a string.
volts	Volts of transceiver; a string or N/A.
temperature	Temperature of transceiver; a string or N/A.
approved	Approval status of the transceiver; one of *approved, accepted, unapproved, Unsuport, Restrict, No Device*.

IP Interface

The following IP interface URIs are available:

- /nos/api/cfg/ip_interface GET
- /nos/api/cfg/ip_interface/<*if_name*> GET, PUT

The following IP interface commands are available:

- Get IP Properties of All Interfaces
- Get IP Interface Properties
- Update IP Interface Properties

Get IP Properties of All Interfaces

Gets IP properties of all interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/ip_interface
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	"ipv6_prefix_len": 64,
	"ipv6_addr": "5001::1",
	"ip_addr": "11.11.11.1",
	"bridge_port": "no",
	"if_name": "Ethernet1/1",
	"mtu": 1500,
	"vrf_name": "default",
	"admin_state": "down",
	"ip_prefix_len": 24
	}</pre>

where:

Element	Description
`ipv6_prefix_len`	The IPv6 network prefix.
`ipv6_addr`	The IPv6 address.
`if_name`	IP interface name (string). **Note:** The interface must exist.
`bridge_port`	Whether or not the port is a bridge port; one of *yes, no*. Default value: *yes*.
`mtu`	The maximum transmission unit, in bytes; an integer from *64-9216*. Default value: *1500*.
`ip_addr`	IP address for the interface.
`ip_prefix_len`	IP address mask; a positive integer from *0-128*.
`vrf_name`	The name of the VRF to which the interface belongs. **Note:** The named VRF must exist.
`admin_state`	The admin status; one of *up, down*.

Get IP Interface Properties

Gets IP properties of one interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/ip_interface/<*ip_if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	    "ipv6_prefix_len": 64,
	    "ipv6_addr": "5001::1",
	    "ip_addr": "11.11.11.1",
	    "bridge_port": "no",
	    "if_name": "Ethernet1/1",
	    "mtu": 1500,
	    "vrf_name": "default",
	    "admin_state": "down",
	    "ip_prefix_len": 24
	}</pre>

where:

Element	Description
`ipv6_prefix_len`	The IPv6 network prefix.
`ipv6_addr`	The IPv6 address.
`if_name`	IP interface name (string). **Note:** The interface must exist.
`bridge_port`	Whether or not the port is a bridge port; one of *yes, no*. Default value: *yes*.
`mtu`	The maximum transmission unit, in bytes; an integer from *64-9216*. Default value: *1500*.
`ip_addr`	IP address for the interface.
`ip_prefix_len`	IP address mask; a positive integer from *1-32*.
`vrf_name`	The name of the VRF to which the interface belongs. **Note:** The named VRF must exist.
`admin_state`	The admin status; one of *up, down*.

Update IP Interface Properties

Updates the IP properties of one interface.

Note: If an element is not specified in a PUT request, no update for that element will be performed.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ip_interface/<*ip_if_name*>

Request Body
 	
(JSON)
	<pre>{
	"ipv6_prefix_len": 64,
	"ipv6_addr": "4001::1",
	"ip_addr": "12.11.11.1",
	"bridge_port": "no",
	"if_name": "Ethernet1/1",
	"mtu": 1500,
	"vrf_name": "default",
	"admin_state": "down",
	"ip_prefix_len": 24
	}</pre>

where:

Element	Description
`ipv6_prefix_len`	The IPv6 network prefix.
`ipv6_addr`	The IPv6 address.
`if_name`	IP interface name (string).The interface must exist.
`bridge_port`	Whether or not the port is a bridge port; one of *yes, no*. Default value: *yes*.
`mtu`	The maximum transmission unit, in bytes; an integer from *64-9216*. Default value: *1500*.
`ip_addr`	IP address for the interface.
`ip_prefix_len`	IP address mask; a positive integer from *1-32*.
`vrf_name`	The name of the VRF to which the interface belongs. **Note:** The named VRF must exist.
`admin_state`	The admin status; one of *up, down*.

Response

Response Body
 	
(JSON)
	{
	"ipv6_prefix_len": 64,
	"ipv6_addr": "4001::1",
	"ip_addr": "12.11.11.1",
	"bridge_port": "no",
	"if_name": "Ethernet1/1",
	"mtu": 1500,
	"vrf_name": "default",
	"admin_state": "down",
	"ip_prefix_len": 24
	}

LACP

The following LACP URI is available:

- /nos/api/cfg/lacp GET, PUT

The following LACP commands are available:

- Get LACP System Properties
- Update LACP System Properties

Get LACP System Properties

Gets the LACP properties of the system.

Request

Method Type	GET
Request URI	/nos/api/cfg/lacp
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"sys_prio": "*<sys_prio>*",
	"max_bundle": "*<max_bundle>*",
	"interfaces": [
	{
	"if_name": "*<if_name>*",
	"lag_mode": "*<lag_mode>*",
	"lacp_prio": "*<lacp_prio>*",
	"lacp_timeout": "*<lacp_timeout>*"
	}
	]
	}

where:

Element	Description
`sys_prio`	LACP system priority.; a positive integer from *1-65535*. Default value: *32768*.
`max_bundle`	The supported maximum number of links per LAG; a positive integer.
`if_name`	Ethernet interface name (string). **Note:** The interface must exist.
`lag_mode`	LAG mode; one of *lacp_active, lacp_passive, no_lacp*.
`lacp_prio`	LACP priority for the physical port; a positive integer from *1-65535*. Default value: *32768*.
`lacp_timeout`	LACP timeout for the physical port; one of *short, long*. Default value: *long*.

Update LACP System Properties

Updates the LACP properties of the system.

Request

Method Type	PUT
Request URI	/nos/api/cfg/lacp

Request Body
 	
(JSON)
	{
	  "sys_prio": "*<sys_prio>*",
	}

where:

Element	Description
`sys_prio`	LACP system priority; a positive integer from *1-65535*. Default value: *32768*.

Response

Response Body
 	
(JSON)
	{
	  "sys_prio": "*<sys_prio>*",
	}

where:

Element	Description
`sys_prio`	LACP system priority; a positive integer from *1-65535*. Default value: *32768*.

LAG

The following LAG-related URIs are available:

- /nos/api/cfg/lag — GET, POST
- /nos/api/cfg/lag/<*lag_id*> — GET, PUT, DELETE
- /nos/api/cfg/lag/load_balance> — GET, PUT

The following LAG commands are available:

- Get All LAGs
- Create LAG
- Get LAG Properties
- Update LAG
- Get LAG Load Balance Settings
- Update LAG Load Balance Settings
- Delete LAG

Get All LAGs

Gets properties of all LAGs.

Request

Method Type	GET
Request URI	/nos/api/cfg/lag
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "lag_name": "*<lag_name>*",
	  "lag_id": "*<lag_id>*",
	  "interfaces": [
	  {
	  "if_name": "*<if_name>*",
	  "lag_mode": "*<lag_mode>*",
	  "lacp_prio": "*<lacp_prio>*",
	  "lacp_timeout": "*<lacp_timeout>*"
	  }
	  ],
	  "suspend_individual": "*<status>*",
	  "min_links": "*<min_links>*",
	  }
	]

where:

Element	Description
`lag_name`	The name of the LAG (string).
`lag_id`	LAG identifier; an integer from *1-65535*.
`interfaces`	Physical interface members of the LAG; an integer from *1-32*.
`if_name`	Ethernet interface name. **Note:** The interface must exist.
`lag_mode`	LAG mode; one of *lacp_active, lacp_passive, no_lacp*.
`lacp_prio`	LACP priority for the physical port; an integer from *1-65535*. Default value: *32768*.
`lacp_timeout`	LACP timeout for the physical port; one of *short, long*. Default value: *long*.

Element	Description
`suspend_ individual`	If the LAG does not get the LACP BPUD from peer ports the port aggregation, the result is one of the following: • *suspended*: LACP on the ports is suspended rather than put into individual state • *individual*: LAG on the ports is put into individual state Default value: *suspended.*
`min_links`	LACP minimum links number; an integer from *1-32*. Default value: *1*.

Create LAG

Creates a LAG.

Request

Method Type	POST
Request URI	/nos/api/cfg/lag

Request Body
 	
(JSON)
	{
	"lag_id": "*<lag_id>*",
	"interfaces": [
	{
	"if_name": "*<if_name>*",
	"lag_mode": "*<lag_mode>*",
	"lacp_prio": "*<lacp_prio>*",
	"lacp_timeout": "*<lacp_timeout>*"
	}
	]
	}

where:

Element	Description
`lag_id`	LAG identifier; a positive integer from *1-65535*.
`interfaces`	Physical interface members of the LAG. Up to 32 interfaces can be added.
`if_name`	Ethernet interface name (string). **Note:** The interface must exist.
`lag_mode`	LAG mode; one of *lacp_active, lacp_passive, no_lacp*.
`lacp_prio`	(Optional) LACP priority for the physical port; a positive integer from *1-65535*. Default value: *32768*.
`lacp_timeout`	(Optional) LACP timeout for the physical port; one of *short, long*. Default value: *long*.

Response

Response Body
 	
(JSON)
	{
	"lag_id": "*<lag_id>*",
	"lag_name": "*<lag_name>*",
	"interfaces": [
	{
	"if_name": "*<if_name>*",
	"lag_mode": "*<lag_mode>*",
	"lacp_prio": "*<lacp_prio>*",
	"lacp_timeout": "*<lacp_timeout>*"
	}
	]
	}

where:

Element	Description
`lag_id`	LAG identifier; a positive integer from *1-65535*.
`interfaces`	Physical interface members of the LAG. Up to 32 interfaces can be added.
`if_name`	Ethernet interface name (string). **Note:** The interface must exist.
`lag_mode`	LAG mode; one of *lacp_active, lacp_passive, no_lacp*.
`lacp_prio`	(Optional) LACP priority for the physical port; a positive integer from *1-65535*. Default value: *32768*.
`lacp_timeout`	(Optional) LACP timeout for the physical port; one of *short, long*. Default value: *long*.
`lag_id`	LAG identifier; a positive integer from *1-65535*.

Get LAG Properties

Gets properties of the specified LAG.

Request

Method Type	GET
Request URI	/nos/api/cfg/lag/*<lag_id>*
Request Body (JSON)	

where:

Element	Description
lag_id	LAG identifier; an integer from 1-65535.

Response

Response Body

(JSON)

```
[
  {
    "lag_name":  "<lag_name>",
    "lag_id":  "<lag_id>",
    "interfaces": [
      {
        "if_name":  "<if_name>",
        "lag_mode":  "<lag_mode>",
        "lacp_prio":  "<lacp_prio>",
        "lacp_timeout":  "<lacp_timeout>"
      }
    ],
    "suspend_individual":  "<status>",
    "min_links":  "<min_links>",
  }
]
```

where:

Element	Description
`lag_name`	The name of the LAG (string).
`lag_id`	LAG identifier; an integer from *1-65535*.
`interfaces`	Physical interface members of the LAG; an integer from *1-32*.
`if_name`	Ethernet interface name. **Note:** The interface must exist.
`lag_mode`	LAG mode; one of *lacp_active, lacp_passive, no_lacp*.

Element	Description
`lacp_prio`	LACP priority for the physical port; an integer from *1-65535*. Default value: *32768*.
`lacp_timeout`	LACP timeout for the physical port; one of *short, long*. Default value: *long*.
`suspend_individual`	If the LAG does not get the LACP BPUD from peer ports the port aggregation, the result is one of the following: • *suspended*: LACP on the ports is suspended rather than put into individual state • *individual*: LAG on the ports is put into individual state Default value: *suspended.*
`min_links`	LACP minimum links number; an integer from *1-32*. Default value: *1*.

Update LAG

Note: If an element is not specified in a PUT request, no update for that element will be performed.

Request

Method Type	PUT
Request URI	/nos/api/cfg/lag/<*lag_id*>

Request Body
 	
(JSON)
	[
	{
	"lag_id": "<*lag_id*>",
	"interfaces": [
	{
	"if_name": "<*if_name*>",
	"lag_mode": "<*lag_mode*>",
	"lacp_prio": "<*lacp_prio*>",
	"lacp_timeout": "<*lacp_timeout*>"
	}
	],
	"suspend_individual": "<*status*>",
	"min_links": "<*min_links*>",
	}
	]

where:

Element	Description
`lag_id`	LAG identifier; an integer from *1-65535*.
`interfaces`	(Optional) Physical interface members of the LAG; an integer from *1-32*.
`if_name`	Ethernet interface name. **Note:** The interface must exist.
`lag_mode`	LAG mode; one of *lacp_active, lacp_passive, no_lacp*.
`lacp_prio`	(Optional) LACP priority for the physical port; an integer from *1-65535.* Default value: *32768*.
`lacp_timeout`	(Optional) LACP timeout for the physical port; one of *short, long*. Default value: *long*.
`suspend_individual`	If the LAG does not get the LACP BPUD from peer ports the port aggregation, the result is one of the following: • *suspended*: LACP on the ports is suspended rather than put into individual state • *individual*: LAG on the ports is put into individual state Default value: *suspended.*
`min_links`	LACP minimum links number; an integer from *1-32*. Default value: *1*.

Response

Response Body
 	
(JSON)
	[
	  {
	  "lag_id": "*<lag_id>*",
	  "lag_name": "*<lag_name>*",
	  "interfaces": [
	  {
	  "if_name": "*<if_name>*",
	  "lag_mode": "*<lag_mode>*",
	  "lacp_prio": "*<lacp_prio>*",
	  "lacp_timeout": "*<lacp_timeout>*"
	  }
	  ],
	  "suspend_individual": "*<status>*",
	  "min_links": "*<min_links>*",
	  }
	]

where:

Element	Description
`lag_id`	LAG identifier; an integer from *1-65535*.
`interfaces`	(Optional) Physical interface members of the LAG; an integer from *1-32*.
`if_name`	Ethernet interface name. **Note:** The interface must exist.
`lag_mode`	LAG mode; one of *lacp_active, lacp_passive, no_lacp*.
`lacp_prio`	(Optional) LACP priority for the physical port; an integer from *1-65535.* Default value: *32768*.
`lacp_timeout`	(Optional) LACP timeout for the physical port; one of *short, long*. Default value: *long*.
`suspend_individual`	If the LAG does not get the LACP BPUD from peer ports the port aggregation, the result is one of the following: • *suspended*: LACP on the ports is suspended rather than put into individual state • *individual*: LAG on the ports is put into individual state Default value: *suspended.*
`min_links`	LACP minimum links number; an integer from *1-32*. Default value: *1*.
`lag_id`	LAG identifier; an integer from *1-65535*.

Get LAG Load Balance Settings

Gets the load balance properties for port aggregations.

Request

Method Type	GET
Request URI	/nos/api/cfg/lag/load_balance
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"destination-ip" : "*<destination-ip>*"
	"destination-mac" : "*<destination-mac>*"
	"destination-port" : "*<destination-port>*"
	"source-dest-ip" : "*<source-dest-ip>*"
	"source-dest-mac" : "*<source-dest-mac>*"
	"source-dest-port" : "*<source-dest-port>*"
	"source-interface" : "*<source-interface>*"
	"source-ip" : "*<source-ip>*"
	"source-mac" : "*<source-mac>*"
	"source-port" : "*<source-port>*"
	}

where:

Element	Description
`destination-ip`	Load distribution on the destination IP address.
`destination-mac`	Load distribution on the destination MAC address.
`destination-port`	Load distribution on the destination TCP/UDP port.
`source-dest-ip`	Load distribution on the source and destination IP address.
`source-dest-mac`	Load distribution on the source and destination MAC address.
`source-dest-port`	Load distribution on the source and destination TCP/UDP port.
`source-interface`	Load distribution on the source ethernet interface.
`source-ip`	Load distribution on the source IP address.

Element	Description
`source-mac`	Load distribution on the source MAC address.
`source-port`	Load distribution on the source TCP/UDP port.

Update LAG Load Balance Settings

Updates the load balance properties for port aggregations.

Request

Method Type	PUT
Request URI	/nos/api/cfg/lag/load_balance

Request Body
 	
(JSON)
	{
	"destination-ip" : "*<destination-ip>*"
	"destination-mac" : "*<destination-mac>*"
	"destination-port" : "*<destination-port>*"
	"source-dest-ip" : "*<source-dest-ip>*"
	"source-dest-mac" : "*<source-dest-mac>*"
	"source-dest-port" : "*<source-dest-port>*"
	"source-interface" : "*<source-interface>*"
	"source-ip" : "*<source-ip>*"
	"source-mac" : "*<source-mac>*"
	"source-port" : "*<source-port>*"
	}

where:

Element	Description
`destination-ip`	Load distribution on the destination IP address.
`destination-mac`	Load distribution on the destination MAC address.
`destination-port`	Load distribution on the destination TCP/UDP port.
`source-dest-ip`	Load distribution on the source and destination IP address.
`source-dest-mac`	Load distribution on the source and destination MAC address.
`source-dest-port`	Load distribution on the source and destination TCP/UDP port.
`source-interface`	Load distribution on the source ethernet interface.
`source-ip`	Load distribution on the source IP address.
`source-mac`	Load distribution on the source MAC address.
`source-port`	Load distribution on the source TCP/UDP port.

Response

Response Body
 	
(JSON)
	<pre>{
	  "destination-port": "no",
	  "source-dest-port": "yes",
	  "source-ip": "no",
	  "source-dest-ip": "no",
	  "destination-mac": "no",
	  "source-mac": "no",
	  "destination-ip": "no",
	  "source-interface": "no",
	  "source-port": "no",
	  "source-dest-mac": "no"
	}</pre>

where:

Element	Description
`destination-ip`	Load distribution on the destination IP address.
`destination-mac`	Load distribution on the destination MAC address.
`destination-port`	Load distribution on the destination TCP/UDP port.
`source-dest-ip`	Load distribution on the source and destination IP address.
`source-dest-mac`	Load distribution on the source and destination MAC address.
`source-dest-port`	Load distribution on the source and destination TCP/UDP port.
`source-interface`	Load distribution on the source ethernet interface.
`source-ip`	Load distribution on the source IP address.
`source-mac`	Load distribution on the source MAC address.
`source-port`	Load distribution on the source TCP/UDP port.

Delete LAG

Deletes a LAG.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/lag/<*lag_id*>
Request Body (JSON)	

where:

Element	Description
lag_id	LAG identifier; a positive integer from 1-65535.

Note: If there is no lag_id (*lag_id*=`None` or specified *lag_id*=`All`), all user-created LAGs will be deleted.

LDAP

The following Lightweight Directory Access Protocol (LDAP) URIs are available:

- /nos/api/cfg/ldap GET, PUT
- /nos/api/cfg/ldap/profiles GET, POST, DELETE
- /nos/api/cfg/ldap/groups GET, POST, DELETE

The following LDAP commands are available:

- Get LDAP Configuration
- Update LDAP Configuration
- Get LDAP Profile Configuration
- Add LDAP Profile
- Delete LDAP Profile
- Get LDAP Server Group Information
- Add LDAP Server Group
- Delete LDAP Server Group

Get LDAP Configuration

Gets the current LDAP configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/ldap
Request Body (JSON)	

Response

Response Body (JSON)	{
	"status":{enable|disable},
	"global_pki":{*pki_name*},
	"global_authorization":{bitmap|rbac},
	"global_retransmit":{*retransmit*},
	"global_timeout":{*timeout*}
	}

where:

Element	Description
`status`	The global status of the LDAP service on the switch; one of *enable, disable.*
`global_pki`	The name of the PKI profile used by LDAP (string).
`global_authorization`	The global LDAP authentication method; one of: • *bitmap* for 8-byte permission bitmaps • *rbac* for Role-Based Access Control (RBAC)
`global_retransmit`	The global LDAP server retransmit count; an integer from *1-5.*
`global_timeout`	The global LDAP server timeout, in seconds; an integer from *1-60.*

Update LDAP Configuration

Updates the LDAP configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ldap

Request Body (JSON)	{
	"status":{enable|disable},
	"global_pki":{*pki_name*},
	"global_authorization":{bitmap|rbac},
	"global_retransmit":{*retransmit*},
	"global_timeout":{*timeout*}
	}

where:

Element	Description
`status`	The global status of the LDAP service on the switch; one of *enable, disable.*
`global_pki`	The name of the PKI profile used by LDAP (string).
`global_ authorization`	The global LDAP authentication method; one of: • *bitmap* for 8-byte permission bitmaps • *rbac* for Role-Based Access Control (RBAC)
`global_ retransmit`	The global LDAP server retransmit count; an integer from *1-5.*
`global_ timeout`	The global LDAP server timeout, in seconds; an integer from *1-60.*

Response

Response Body (JSON)	{
	"status":{enable|disable},
	"global_pki":{*pki_name*},
	"global_authorization":{bitmap|rbac},
	"global_retransmit":{*retransmit*},
	"global_timeout":{*timeout*}
	}

where:

Element	Description
`status`	The global status of the LDAP service on the switch; one of *enable, disable.*
`global_pki`	The name of the PKI profile used by LDAP (string).

Element	Description
`global_ authorization`	The global LDAP authentication method; one of: • *bitmap* for 8-byte permission bitmaps • *rbac* for Role-Based Access Control (RBAC)
`global_ retransmit`	The global LDAP server retransmit count; an integer from *1-5.*
`global_ timeout`	The global LDAP server timeout, in seconds; an integer from *1-60.*

Get LDAP Profile Configuration

Gets the configuration of a specific LDAP profile.

Request

Method Type	GET
Request URI	/nos/api/cfg/ldap/profiles/<*profile_name*>
Request Body (JSON)	

where:

Element	Description
profile_name	The name of the LDAP profile (string).

Response

Response Body (JSON)	[
	  {
	  "profile_name":{*profile_name*},
	  "host":{*ip_address*},
	  "port":{*port*},
	  "base_dn":{*base_dn*},
	  "bind_mode":{prompted|predefined},
	  "security":{ldaps (ignore)|startTLS (ignore)|clear},
	  "retransmit":{*retransmit*},
	  "timeout":{*timeout*},
	  "authorization":{bitmap|rbac},
	  "attribute_group":{*attribute_group*},
	  "attribute_permission_name":{*attribute_permission_name*},
	  "attribute_permission_admin_bitmap":
	  {*attribute_permission_admin_bitmap*},
	  "attribute_permission_oper_bitmap":
	  {*attribute_permission_oper_bitmap*},
	  "attribute_permission_deny_bitmap":
	  {*attribute_permission_deny_bitmap*},
	  "attribute_permission_admin_role":
	  {*attribute_permission_admin_role*},
	  "attribute_permission_oper_role":
	  {*attribute_permission_oper_role*},
	  "attribute_permission_deny_role":
	  {*attribute_permission_deny_role*},
	  "attribute_username":{*attribute_username*},
	  "predefined_credential_dn":{*predefined_credential_dn*},
	  "predefined_credential_key":{*predefined_credential_key*},
	  "credential_key_form":{0|7},
	  "group_filter":{*group_filter*},
	  "pki_name":{pki_name}
	  }
	]

where:

Element	Description
`profile_name`	The name of the LDAP profile (string).
`host`	The IP address of the LDAP server.
`port`	The TCP port for sending messages to the LDAP server; an integer from *1-65535.*
`base_dn`	The LDAP based Domain Name (DN); a string.
`bind_mode`	The LDAP binding method; one of *prompted, predefined.*
`security`	The LDAP transmit mode and security option; one of: • *LDAPS (ignore)* • *startTLS (ignore)* • *clear*
`retransmit`	The LDAP retransmit count; an integer from *1-5.*

Element	Description
`timeout`	The LDAP server connection timeout period, in seconds; an integer from *1-60.*
`authorization`	The LDAP authorization method; one of: • *bitmap* for 8-byte permission bitmaps • *rbac* for Role-Based Access Control (RBAC)
`attribute_group`	The name of the custom LDAP attribute group (string).
`attribute_permission_name`	The custom LDAP attribute permission name (string).
`attribute_permission_admin_bitmap`	The custom LDAP attribute permission administrative bitmap (string).
`predefined_credential_dn`	The Domain Name for binding with the LDAP server (string).
`attribute_permission_admin_role`	The custom LDAP attribute permission administrator role (string).
`attribute_permission_oper_bitmap`	The custom LDAP attribute permission operative bitmap (string).
`attribute_permission_deny_bitmap`	The custom LDAP attribute permission denial bitmap (string).
`credential_key_form`	The LDAP authentication key encryption method; one of: • *0* - clear text • *7* - encrypted
`predefined_credential_key`	The password for the Domain Name (string).
`attribute_username`	The custom LDAP attribute username (string).
`attribute_permission_oper_role`	The custom LDAP attribute permission operator role (string).
`attribute_permission_deny_role`	The custom LDAP attribute deny operator role.
`group_filter`	The filter when performing group searches (string).
`pki_name`	The name of the PKI profile used by LDAP (string).

Add LDAP Profile

Configures a new LDAP profile.

Request

Method Type	POST
Request URI	/nos/api/cfg/ldap/profiles

Request Body (JSON)	{
	"profile_name":{*profile_name*},
	"host":{*ip_address*},
	"port":{*port*},
	"base_dn":{*base_dn*},
	"bind_mode":{prompted|predefined},
	"security":{ldaps (ignore)|startTLS (ignore)|clear},
	"retransmit":{*retransmit*},
	"timeout":{*timeout*},
	"authorization":{bitmap|rbac},
	"attribute_group":{*attribute_group*},
	"attribute_permission_name":{*attribute_permission_name*},
	"attribute_permission_admin_bitmap":
	{*attribute_permission_admin_bitmap*},
	"attribute_permission_oper_bitmap":
	{*attribute_permission_oper_bitmap*},
	"attribute_permission_deny_bitmap":
	{*attribute_permission_deny_bitmap*},
	"attribute_permission_admin_role":
	{*attribute_permission_admin_role*},
	"attribute_permission_oper_role":
	{*attribute_permission_oper_role*},
	"attribute_permission_deny_role":
	{*attribute_permission_deny_role*},
	"attribute_username":{*attribute_username*},
	"predefined_credential_dn":{*predefined_credential_dn*},
	"predefined_credential_key":{*predefined_credential_key*},
	"credential_key_form":{0|7},
	"group_filter":{*group_filter*},
	"pki_name":{pki_name}
	}

where:

Element	Description
`profile_name`	The name of the LDAP profile (string).
`host`	The IP address of the LDAP server.
`port`	The TCP port for sending messages to the LDAP server; a string from *1-65535.*
`base_dn`	The LDAP based Domain Name (DN) (string).
`bind_mode`	The LDAP binding method; one of: *prompted, predefined.*

Element	Description
`security`	The LDAP transmit mode and security option; one of: • *LDAPS (ignore)* • *startTLS (ignore)* • *clear*
`retransmit`	The LDAP retransmit count; an integer from *1-5.*
`timeout`	The LDAP server connection timeout period, in seconds; an integer from *1-60.*
`authorization`	The LDAP authorization method; one of: • *bitmap* for 8-byte permission bitmaps • *rbac* for Role-Based Access Control (RBAC)
`attribute_group`	The name of the custom LDAP attribute group (string).
`attribute_permission_name`	The custom LDAP attribute permission name (string).
`attribute_permission_admin_bitmap`	The custom LDAP attribute permission administrative bitmap (string).
`predefined_credential_dn`	The Domain Name for binding with the LDAP server (string).
`attribute_permission_admin_role`	The custom LDAP attribute permission administrator role (string).
`attribute_permission_oper_bitmap`	The custom LDAP attribute permission operative bitmap (string).
`attribute_permission_deny_bitmap`	The custom LDAP attribute permission denial bitmap (string).
`credential_key_form`	The LDAP authentication key encryption method; one of: • *0* - clear text • *7* - encrypted
`predefined_credential_key`	The password for the Domain Name (string).
`attribute_username`	The custom LDAP attribute username (string).
`attribute_permission_oper_role`	The custom LDAP attribute permission operator role (string).

Element	Description
`attribute_ permission_deny_ role`	The custom LDAP attribute deny operator role.
`group_filter`	The filter when performing group searches (string).
`pki_name`	The name of the PKI profile used by LDAP (string).

Response

Response Body (JSON)

```
{
    "profile_name":{profile_name},
    "host":{ip_address},
    "port":{port},
    "base_dn":{base_dn},
    "bind_mode":{prompted|predefined},
    "security":{ldaps (ignore)|startTLS (ignore)|clear},
    "retransmit":{retransmit},
    "timeout":{timeout},
    "authorization":{bitmap|rbac},
    "attribute_group":{attribute_group},
    "attribute_permission_name":{attribute_permission_name},
    "attribute_permission_admin_bitmap":
              {attribute_permission_admin_bitmap},
    "attribute_permission_oper_bitmap":
              {attribute_permission_oper_bitmap},
    "attribute_permission_deny_bitmap":
              {attribute_permission_deny_bitmap},
    "attribute_permission_admin_role":
              {attribute_permission_admin_role},
    "attribute_permission_oper_role":
              {attribute_permission_oper_role},
    "attribute_permission_deny_role":
              {attribute_permission_deny_role},
    "attribute_username":{attribute_username},
    "predefined_credential_dn":{predefined_credential_dn},
    "predefined_credential_key":{predefined_credential_key},
    "credential_key_form":{0|7},
    "group_filter":{group_filter},
    "pki_name":{pki_name}
}
```

where:

Element	Description
`profile_name`	The name of the LDAP profile (string).
`host`	The IP address of the LDAP server.
`port`	The TCP port for sending messages to the LDAP server; an integer from *1-65535.*
`base_dn`	The LDAP based Domain Name (DN) (string).

Element	Description
`bind_mode`	The LDAP binding method; one of *prompted, predefined.*
`security`	The LDAP transmit mode and security option; one of: • *LDAPS (ignore)* • *startTLS (ignore)* • *clear*
`retransmit`	The LDAP retransmit count; an integer from *1-5.*
`timeout`	The LDAP server connection timeout period, in seconds; an integer from *1-60.*
`authorization`	The LDAP authorization method; one of: • *bitmap* for 8-byte permission bitmaps • *rbac* for Role-Based Access Control (RBAC)
`attribute_group`	The name of the custom LDAP attribute group (string).
`attribute_ permission_name`	The custom LDAP attribute permission name (string).
`attribute_permission_ admin_bitmap`	The custom LDAP attribute permission administrative bitmap (string).
`predefined_ credential_dn`	The Domain Name for binding with the LDAP server (string).
`attribute_ permission_admin_role`	The custom LDAP attribute permission administrator role (string).
`attribute_ permission_oper_ bitmap`	The custom LDAP attribute permission operative bitmap (string).
`attribute_ permission_deny_ bitmap`	The custom LDAP attribute permission denial bitmap (string).
`credential_key_ form`	The LDAP authentication key encryption method; one of: • *0* - clear text • *7* - encrypted
`predefined_ credential_key`	The password for the Domain Name (string).
`attribute_username`	The custom LDAP attribute username (string).
`attribute_ permission_oper_role`	The custom LDAP attribute permission operator role (string).

Element	Description
`attribute_ permission_deny_role`	The custom LDAP attribute deny operator role.
`group_filter`	The filter when performing group searches (string).
`pki_name`	The name of the PKI profile used by LDAP (string).

Delete LDAP Profile

Deletes a configured LDAP profile.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/ldap/profiles/<*profile_name*>
Request Body (JSON)	

where:

Element	Description
profile_name	The name of the LDAP profile (string).

Get LDAP Server Group Information

Gets LDAP group information.

Request

Method Type	GET
Request URI	/nos/api/cfg/ldap/groups/<*group_name*>
Request Body (JSON)	

where:

Element	Description
group_name	The name of the LDAP server group (string). **Note:** If this variable is not provided, then the command returns information about all configured LDAP server groups.

Response

Response Body (JSON)	<pre>[
	  {
	    "group_name":"{group_name}",
	    "vrf_name":"{vrf_name}",
	    "profiles":
	    [
	      {
	        "profile_name":{profile_name},
	        "host":{ip_address},
	        "port":{port},
	        "base_dn":{base_dn},
	        "bind_mode":{prompted|predefined},
	        "security":{ldaps (ignore)|startTLS (ignore)|clear},
	        "retransmit":{retransmit},
	        "timeout":{timeout},
	        "authorization":{bitmap|rbac},
	        "attribute_group":{attribute_group},
	        "attribute_permission_name":{attribute_permission_name},
	        "attribute_permission_admin_bitmap":
	                  {attribute_permission_admin_bitmap},
	        "attribute_permission_oper_bitmap":
	                  {attribute_permission_oper_bitmap},
	        "attribute_permission_deny_bitmap":
	                  {attribute_permission_deny_bitmap},
	        "attribute_permission_admin_role":
	                  {attribute_permission_admin_role},
	        "attribute_permission_oper_role":
	                  {attribute_permission_oper_role},
	        "attribute_permission_deny_role":
	                  {attribute_permission_deny_role},
	        "attribute_username":{attribute_username},
	        "predefined_credential_dn":{predefined_credential_dn},
	        "predefined_credential_key":{predefined_credential_key},
	        "credential_key_form":{0|7},
	        "group_filter":{group_filter},
	        "pki_name":{pki_name}
	      }
	    ]
	  }
	]</pre>

where:

Element	Description
`group_name`	The name of the LDAP server group (string).
`vrf_name`	The VRF instance for the LDAP server group.
`profiles`	The list of LDAP profiles.
`profile_name`	The name of the LDAP profile (string).
`host`	The IP address of the LDAP server.
`port`	The TCP port for sending messages to the LDAP server; an integer from *1-65535.*
`base_dn`	The LDAP based Domain Name (DN) (string).

Element	Description
`bind_mode`	The LDAP binding method; one of: *prompted, predefined.*
`security`	The LDAP transmit mode and security option; one of: • *LDAPS (ignore)* • *startTLS (ignore)* • *clear*
`retransmit`	The LDAP retransmit count; an integer from *1-5.*
`timeout`	The LDAP server connection timeout period, in seconds; an integer from *1-60.*
`authorization`	The LDAP authorization method; one of: • *bitmap* for 8-byte permission bitmaps • *rbac* for Role-Based Access Control (RBAC)
`attribute_group`	The name of the custom LDAP attribute group (string).
`attribute_ permission_name`	The custom LDAP attribute permission name (string).
`attribute_ permission_admin_ bitmap`	The custom LDAP attribute permission administrative bitmap (string).
`predefined_ credential_dn`	The Domain Name for binding with the LDAP server (string).
`attribute_ permission_admin_ role`	The custom LDAP attribute permission administrator role (string).
`attribute_ permission_oper_ bitmap`	The custom LDAP attribute permission operative bitmap (string).
`attribute_ permission_deny_ bitmap`	The custom LDAP attribute permission denial bitmap (string).
`credential_key_form`	The LDAP authentication key encryption method; one of: • *0* - clear text • *7* - encrypted
`predefined_credentia_ key`	The password for the Domain Name (string).
`attribute_username`	The custom LDAP attribute username (string).

Element	Description
`attribute_permission_oper_role`	The custom LDAP attribute permission operator role (string).
`attribute_permission_deny_role`	The custom LDAP attribute deny operator role (string).
`group_filter`	The filter when performing group searches (string).
`pki_name`	The name of the PKI profile used by LDAP (string).

Add LDAP Server Group

Configures a new LDAP server group.

Request

Method Type	POST
Request URI	/nos/api/cfg/ldap/groups

Request Body (JSON)	{
	"group_name":"{*group_name*}",
	"vrf_name":"{*vrf_name*}",
	"profile_name":
	[
	{*profile_name*}
	]
	}

where:

Element	Description
`group_name`	The name of the LDAP server group (string).
`vrf_name`	The VRF instance for the LDAP server group. Valid value: the VRF instance name.
`profile_name`	The list of LDAP profiles.
profile_name	The name of the LDAP profile (string).

Response

Response Body (JSON)

```
[
  {
    "group_name":"{group_name}",
    "vrf_name":"{vrf_name}",
    "profiles":
    [
      {
        "profile_name":{profile_name},
        "host":{ip_address},
        "port":{port},
        "base_dn":{base_dn},
        "bind_mode":{prompted|predefined},
        "security":{ldaps (ignore)|startTLS (ignore)|clear},
        "retransmit":{retransmit},
        "timeout":{timeout},
        "authorization":{bitmap|rbac},
        "attribute_group":{attribute_group},
        "attribute_permission_name":{attribute_permission_name},
        "attribute_permission_admin_bitmap":
                    {attribute_permission_admin_bitmap},
        "attribute_permission_oper_bitmap":
                    {attribute_permission_oper_bitmap},
        "attribute_permission_deny_bitmap":
                    {attribute_permission_deny_bitmap},
        "attribute_permission_admin_role":
                    {attribute_permission_admin_role},
        "attribute_permission_oper_role":
                    {attribute_permission_oper_role},
        "attribute_permission_deny_role":
                    {attribute_permission_deny_role},
        "attribute_username":{attribute_username},
        "predefined_credential_dn":{predefined_credential_dn},
        "predefined_credential_key":{predefined_credential_key},
        "credential_key_form":{0|7},
        "group_filter":{group_filter},
        "pki_name":{pki_name}
      }
    ]
  }
]
```

where:

Element	Description
`group_name`	The name of the LDAP server group (string).
`vrf_name`	The VRF instance for the LDAP server group. Valid value: the VRF instance name.
`profiles`	The list of LDAP profiles.
`profile_name`	The name of the LDAP profile (string).
`host`	The IP address of the LDAP server.
`port`	The TCP port for sending messages to the LDAP server; an integer from *1-65535.*

Element	Description
`base_dn`	The LDAP based Domain Name (DN) (string).
`bind_mode`	The LDAP binding method; one of *prompted, predefined*.
`security`	The LDAP transmit mode and security option; one of: • *LDAPS (ignore)* • *startTLS (ignore)* • *clear*
`retransmit`	The LDAP retransmit count; an integer from *1-5*.
`timeout`	The LDAP server connection timeout period, in seconds; an integer from *1-60*.
`authorization`	The LDAP authorization method; one of: • *bitmap* for 8-byte permission bitmaps • *rbac* for Role-Based Access Control (RBAC)
`attribute_group`	The name of the custom LDAP attribute group (string).
`attribute_permission_name`	The custom LDAP attribute permission name (string).
`attribute_permission_admin_bitmap`	The custom LDAP attribute permission administrative bitmap (string).
`predefined_credential_dn`	The Domain Name for binding with the LDAP server (string).
`attribute_permission_admin_role`	The custom LDAP attribute permission administrator role (string).
`attribute_permission_oper_bitmap`	The custom LDAP attribute permission operative bitmap (string).
`attribute_permission_deny_bitmap`	The custom LDAP attribute permission denial bitmap (string).
`credential_key_form`	The LDAP authentication key encryption method; one of: • *0* - clear text • *7* - encrypted
`predefined_credential_key`	The password for the Domain Name (string).
`attribute_username`	The custom LDAP attribute username (string).
`attribute_permission_oper_role`	The custom LDAP attribute permission operator role (string).

Element	Description
`attribute_permission_deny_role`	The custom LDAP attribute deny operator role.
`group_filter`	The filter when performing group searches (string).
`pki_name`	The name of the PKI profile used by LDAP (string).

Delete LDAP Server Group

Deletes a configured LDAP server group.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/ldap/groups/<*group_name*>
Request Body (JSON)	

where:

Element	Description
group_name	The name of the LDAP server group (string).

LLDP

The following LLDP URIs are available:

- /nos/api/cfg/lldp GET, PUT
- /nos/api/cfg/lldp/lldp_interface GET
- /nos/api/cfg/lldp/lldp_interface/<*eth_if_name*> GET, PUT
- /nos/api/cfg/lldp/lldp_interface/statistics/<*eth_if_name*> GET
- /nos/api/cfg/lldp/lldp_interface/neighbor/<*eth_if_name*> GET
- /nos/api/cfg/lldp/lldp_interface/neighbor GET

The following LLDP commands are available:

- Get LLDP System Properties
- Update LLDP System Properties
- Get LLDP Properties for All Interfaces
- Get LLDP Interface Properties
- Update LLDP Interface Properties
- Get LLDP Interface Statistics
- Get LLDP Interface Neighbor Information
- Get LLDP Neighbor Information for All Interfaces

Get LLDP System Properties

Gets global LLDP properties of the system.

Request

Method Type	GET
Request URI	/nos/api/cfg/lldp
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"reinit delay": "*<reinit delay>*",
	"transit interval": "*<transmit interval>*",
	"transmit delay": "*<transmit delay>*"
	}

where:

Element	Description
`reinit delay`	The number of seconds until LLDP re-initialization is attempted on an interface; an integer from *1-10.* Default value: *2* seconds.
`transmit interval`	The time interval, in seconds, between transmissions of LLDP messages; an integer from *5-32768*. Default value: *30* seconds.
`transmit delay`	The number of seconds for transmission delay; an integer from *1-8192*. Default value: *2* seconds.

Update LLDP System Properties

Updates the global LLDP properties of the system.

Request

Method Type	PUT
Request URI	/nos/api/cfg/lldp

Request Body
 	
(JSON)
	{
	"reinit delay": "*<reinit delay>*",
	"transit interval": "*<transmit interval>*",
	"transmit delay": "*<transmit delay>*"
	}

where:

Element	Description
`reinit delay`	The number of seconds until LLDP re-initialization is attempted on an interface; an integer from *1-10.* Default value: *2* seconds.
`transmit interval`	The time interval, in seconds, between transmissions of LLDP messages; an integer from *5-32768*. Default value: *30* seconds.
`transmit delay`	The number of seconds for transmission delay; an integer from *1-8192*. Default value: *2* seconds.

Response

Response Body
 	
(JSON)
	{
	"reinit delay": "*<reinit delay>*",
	"transit interval": "*<transmit interval>*",
	"transmit delay": "*<transmit delay>*"
	}

Get LLDP Properties for All Interfaces

Gets LLDP properties of all interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/lldp/lldp_interface
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "if_name": "<*if_name*>",
	  "ena_lldp_rx": "<*ena_lldp_rx*>",
	  "ena_lldp_tx": "<*ena_lldp_tx*>"
	  }
	]

where:

Element	Description
`if_name`	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
`ena_lldp_rx`	Enables or disables LLDP frame reception on a physical interface; one of *yes*, *no*. Default value: *yes*.
`ena_lldp_tx`	Enables or disable sLLDP frame transmission on a physical interface; one of *yes*, *no*. Default value: *yes*.

Get LLDP Interface Properties

Gets LLDP properties of one interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/lldp/lldp_interface/<*eth_if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  "if_name": "<*if_name*>",
	  "ena_lldp_rx": "<*ena_lldp_rx*>",
	  "ena_lldp_tx": "<*ena_lldp_tx*>"
	}

where:

Element	Description
`if_name`	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
`ena_lldp_rx`	Enables or disables LLDP frame reception on a physical interface; one of *yes*, *no*. Default value: *yes*.
`ena_lldp_tx`	Enables or disable sLLDP frame transmission on a physical interface; one of *yes*, *no*. Default value: *yes*.

Update LLDP Interface Properties

Updates the LLDP properties of one interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/lldp/lldp_interface/<*eth_if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"ena_lldp_rx": "<*ena_lldp_rx*>",
	"ena_lldp_tx": "<*ena_lldp_tx*>"
	}

where:

Element	Description
`if_name`	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
`ena_lldp_rx`	Enables or disables LLDP frame reception on a physical interface; one of *yes, no*. Default value: *yes*.
`ena_lldp_tx`	Enables or disable sLLDP frame transmission on a physical interface; one of *yes, no*. Default value: *yes*.

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"ena_lldp_rx": "<*ena_lldp_rx*>",
	"ena_lldp_tx": "<*ena_lldp_tx*>"
	}

Get LLDP Interface Statistics

Gets LLDP interface statistics per interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/lldp/lldp_interface/statistics/<*eth_if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"total frames": "<*total_frames*>",
	"total tlvs discarded": "<*total_tlvs_discarded*>",
	"total frames transmitted": "<*total_frames_transmitted*>",
	"total errrored frames": "<*total_errrored_frames*>",
	"total frames discarded": "<*total_frames_discarded*>",
	"total entries aged": "<*total_entries_aged*>",
	"total tlvs unrecognized": "<*total_tlvs_unrecognized*>"
	}

where:

Element	Description
`total frames`	The total number of LLDP frames received.
`total tlvs discarded`	The total number of LLDP TLVs discarded.
`total frames transmitted`	The total number of LLDP frames transmitted.
`total errrored frames`	The total number of frames received with errors.
`total frames discarded`	The total number of discarded frames.
`total entries aged`	The total number of entries aged out.
`total tlvs unrecognized`	The total number of unrecognized LLDP TLVs.

Get LLDP Interface Neighbor Information

Gets LLDP interface neighbor information

Request

Method Type	GET
Request URI	/nos/api/cfg/lldp/lldp_interface/neighbor/<*eth_if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"capability": "<*capability*>",
	"rx ttl": "<*rx ttl*>",
	"system name": "<*system name*>",
	"system description": "<*system description*>"
	"system_mac": "<*system mac*>"
	}

where:

Element	Description
`if_name`	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
`capability`	Remote switch capability; one of *(B) – Bridge, (R) – Router.*
`rx ttl`	The TTL.
`system name`	Remote system name.
`system description`	Remote system description.
`system_mac`	Unique system MAC.

Get LLDP Neighbor Information for All Interfaces

Gets LLDP neighbor information for all interfaces

Request

Method Type	GET
Request URI	/nos/api/cfg/lldp/lldp_interface/neighbor
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	{
	"if_name": "*<if_name>*",
	"capability": "*<capability>*",
	"rx ttl": "*<rx ttl>*",
	"system name": "*<system name>*",
	"system description": "*<system description>*"
	}
	]

where:

Element	Description
`if_name`	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
`capability`	Remote switch capability; one of *(B) – Bridge, (R) – Router.*
`rx ttl`	The TTL.
`system name`	Remote system name.
`system description`	Remote system description.

MSTP

The following MSTP URIs are available:

- /nos/api/cfg/mstp GET, PUT
- /nos/api/cfg/mstp_instance GET, POST
- /nos/api/cfg/mstp_instance/<*instance_number*> GET, PUT, DELETE
- /nos/api/cfg/mstp_interface/<*instance_number*>/<*if_name*> GET, PUT

The following MSTP commands are available:

- Get MSTP System Properties
- Update MSTP System Properties
- Get Properties of All MSTP Instances
- Create MSTP Instance
- Get MSTP Instance
- Update MSTP Instance
- Delete MSTP Instance
- Get Interface Properties of an MSTP Instance
- Update Interface Properties of an MSTP Instance

Get MSTP System Properties

Updates global MSTP properties of the system.

Request

Method Type	GET
Request URI	/nos/api/cfg/mstp
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"region_name": "*<region_name>*"
	"revision": "*<revision>*"
	}

where:

Element	Description
`region_name`	Region name; a string up to 32 characters long.
`revision`	Revision number; an integer from *0-65535*.

Update MSTP System Properties

Updates global MSTP properties of the system.

Note: If an element is not specified in a PUT request, no update for that element will be performed.

Request

Method Type	PUT
Request URI	/nos/api/cfg/mstp

Request Body
 	
(JSON)
	{
	"region_name": "*<region_name>*"
	"revision": "*<revision>*"
	}

where:

Element	Description
`region_name`	Region name; a string up to 32 characters long.
`revision`	Revision number; an integer from *0-65535*.

Response

Response Body
 	
(JSON)
	{
	"region_name": "*<region_name>*"
	"revision": "*<revision>*"
	}

Get Properties of All MSTP Instances

Gets properties of all MSTP instances.

Request

Method Type	GET
Request URI	/nos/api/cfg/mstp_instance
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	{
	"instance_id": "<*instance_id*>",
	"instance_prio": "<*instance_prio*>",
	"vlans": [
	{
	"vlan_id": "<*vlan_id*>"
	}
	]
	}
	]

where:

Element	Description
`instance_id`	MST instance ID; an integer from *0-64*. Instance 0 refers to the CIST.
`instance_prio`	Sets the instance bridge priority; an integer from *0-61440*. Default value: *32768*.
`vlans`	Maps a range of VLANs to a multiple spanning tree instance (MSTI); an integer from *1-4094*.

Create MSTP Instance

Creates an MSTP instance.

Request

Method Type	POST
Request URI	/nos/api/cfg/mstp_instance

Request Body
 	
(JSON)
	<pre>{
	  "instance_id": "<*instance_id*>",
	  "instance_prio": "<*instance_prio*>",
	  "vlans": [
	  {
	  "vlan_id": "<*vlan_id*>"
	  }
	  ]
	}</pre>

where:

Element	Description
`instance_id`	MST instance ID; an integer from *0-64*. Instance 0 refers to the CIST.
`instance_prio`	Sets the instance bridge priority; an integer from *0-61440*. Default value: *32768*.
`vlans`	Maps a range of VLANs to a multiple spanning tree instance (MSTI); an integer from *1-4094*.

Response

Response Body
 	
(JSON)
	<pre>{
	  "instance_id": "<*instance_id*>",
	  "instance_prio": "<*instance_prio*>",
	  "vlans": [
	  {
	  "vlan_id": "<*vlan_id*>"
	  }
	  ]
	}</pre>

Get MSTP Instance

Gets properties of an MSTP instance.

Request

Method Type	GET
Request URI	/nos/api/cfg/mstp_instance/<*instance_id*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"instance_id": "<*instance_id*>",
	"instance_prio": "<*instance_prio*>",
	"vlans": [
	{
	"vlan_id": "<*vlan_id*>"
	}
	]
	}

Update MSTP Instance

Updates the properties of an MSTP instance.

Note: If an element is not specified in a PUT request, no update for that element will be performed.

Request

Method Type	PUT
Request URI	/nos/api/cfg/mstp_instance/<*instance_id*>

Request Body
 	
(JSON)
	{
	"instance_id": "<*instance_id*>",
	"instance_prio": "<*instance_prio*>",
	"vlans": [
	{
	"vlan_id": "<*vlan_id*>"
	}
	]
	}

where:

Element	Description
`instance_id`	MST instance ID; an integer from *0-64*. Instance 0 refers to the CIST.
`instance_prio`	Sets the instance bridge priority; an integer from *0-61440*. Default value: *32768*.
`vlans`	Maps a range of VLANs to a multiple spanning tree instance (MSTI); an integer from *1-4094*.

Response

Response Body
 	
(JSON)
	{
	"instance_id": "<*instance_id*>",
	"instance_prio": "<*instance_prio*>",
	"vlans": [
	{
	"vlan_id": "<*vlan_id*>"
	}
	]
	}

Delete MSTP Instance

Deletes an MSTP instance.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/mstp_instance/<*instance_id*>
Request Body (JSON)	

where:

Element	Description
`instance_id`	MST instance ID; an integer from *0-64*. Instance 0 refers to the CIST.

Get Interface Properties of an MSTP Instance

Gets properties of one interface in an MSTP instance.

Request

Method Type	GET
Request URI	/nos/api/cfg/mstp_interface/<*instance_id*>/<*if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  "if_name": "<*if_name*>",
	  "path_cost": "<*path_cost*>",
	  "port_prio": "<*port_prio*>"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`path_cost`	The port path-cost value on the specified MST instance; either an integer from *1-200000000* or *auto* (default) to base the path-cost on port speed.
`port_prio`	The port priority, in increments of 32, on the specified MST instance; a multiple of 32 from *0-224*. Default value: *128*.

Update Interface Properties of an MSTP Instance

Updates the properties of one interface in an MSTP instance.

Request

Method Type	PUT
Request URI	/nos/api/cfg/mstp_interface/{instance_id}/<*if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"path_cost": "<*path_cost*>",
	"port_prio": "<*port_prio*>"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`path_cost`	The port path-cost value on the specified MST instance; either an integer from *1-200000000* or *auto* (default) to base the path-cost on port speed.
`port_prio`	The port priority, in increments of 32, on the specified MST instance; a multiple of 32 from *0-224*. Default value: *128*.

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"path_cost": "<*path_cost*>",
	"port_prio": "<*port_prio*>"
	}

Nexthophealth

The following nexthophealth URI is available:

- /nos/api/cfg/nhophealth POST

The following nextophealth command is available:

- Nexthop Health Check

Nexthop Health Check

Enables or disables the nexthop health check feature by setting the health checking interval value.

Request

Method Type Type	POST
Request URI	/nos/api/cfg/nhophealth

Request Body
 	
(JSON)
	{
	"healthcheck_interval": *<5-60>*
	}

where:

Element	Description
`healthcheck_interval`	(Mandatory) The global nexthop health check interval, in seconds; an integer from *5-60*. Default value: *0* (disabled).

Response

`True` if the operation succeeded; otherwise `False`.

NOS Copy

The following NOS copy URIs are available:

- /nos/api/saveneeded GET
- /nos/api/save/config GET
- /nos/api/reset GET
- /nos/api/download/image POST
- /nos/api/download/config POST
- /nos/api/upload/config POST
- /nos/api/upload/tech_support POST
- /nos/api/download/status/<*content*> GET
- /nos/api/upload/status/<*content*> GET

The following NOS Copy commands are available:

- Determine Whether the Running Configuration Needs to be Saved
- Reset Switch
- Save Configuration
- Download Image to Switch
- Download Configuration to Switch
- Upload Configuration to Server
- Upload Tech Support to Server
- Get Download Transfer Status
- Get Upload Transfer Status

Note: The requests in this section are required for XClarity support.

Determine Whether the Running Configuration Needs to be Saved

Gets whether the running configuration needs to be saved by checking if there is a difference between the configuration that is running versus what is in flash.

Request

Method Type	GET
Request URI	/nos/api/saveneeded
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"saveneeded" : "*<flag>*"
	}

where:

Element	Description
saveneeded	Whether the running configuration matches what is in flash memory; one of *yes, no*.

Reset Switch

Resets the switch.

Request

Method Type	GET
Request URI	/nos/api/reset
Request Body (JSON)	

Response

`True` if the operation succeeded; otherwise `False`.

Save Configuration

Saves the running configuration to flash memory.

Request

Method Type	GET
Request URI	/nos/api/save/config
Request Body (JSON)	

Response

`True` if the operation succeeded; otherwise `False`.

Download Image to Switch

Downloads a boot image to the switch.

Request

Method Type	POST
Request URI	/nos/api/download/image

Request Body
 	
(JSON)
	{
	"protocol":"*<protocol>*",
	"serverip":"*<serverip>*",
	"srcfile":"*<srcfile>*",
	"imgtype":"*<imgtype>*",
	"username":"*<username>*",
	"passwd":"*<passwd>*",
	"vrf_name":"*<vrf_name>*"
	}

where:

Element	Description
protocol	Protocol name; one of *tftp, sftp.*
serverip	Server IP address.
srcfile	Source file; up to 256 characters long.
imgtype	System image type; one of *all, boot, onie, os*.
username	Username for the server. Not required for TFTP.
passwd	Password for the server username. Not required for TFTP.
vrf_name	(Optional) VRF name; an alphabetic string up to 64 characters long.

Response

Response Body
 	
(JSON)
	{
	"status": "*<status>*",
	}

where:

Element	Description
status	Transfer status; one of *transferring, installing, successful, failed*.

Download Configuration to Switch

Downloads a configuration to the switch.

Request

Method Type	POST
Request URI	/nos/api/download/config

Request Body
 	
(JSON)
	{
	"protocol":"*<protocol>*",
	"serverip":"*<serverip>*",
	"srcfile":"*<srcfile>*",
	"dstfile":"*<dstfile>*",
	"username":"*<username>*",
	"passwd":"*<passwd>*",
	"vrf_name":"*<vrf_name>*"
	}

where:

Element	Description
protocol	Protocol name; one of *tftp, sftp*.
serverip	Server IP address.
srcfile	Source file; up to 256 characters long.
dstfile	Destination file; one of *running_config, startup_config*.
username	(Optional) Username for the server.
passwd	(Optional) Password for the server username.
vrf_name	(Optional) VRF name; an alphabetic string up to 64 characters long.

Response

Response Body
 	
(JSON)
	{
	  "status": "*<status>*",
	  "details": "*<details>*",
	  "filename": "*<filename>*"
	}

where:

Element	Description
`status`	Transfer status; one of *transferring, installing, successful, failed*.
`details`	Detailed description of the status; one of: • *Transferring running-config* • *Transferring startup-config* • *Installing image* • *image installation succeeded* • *Copy success* • *VRF vrf_name doesn't exist* • *Another image installation is in progress* • *Host serverip is unreachable* • *ONIE feature in not enabled on this switch* • *File not found* • *SFTP authentication failure* • *image installation failed* • *Copy failed*
`filename`	Configuration filename.

Upload Configuration to Server

Uploads a configuration from the switch to a server.

Request

Method Type	POST
Request URI	/nos/api/upload/config

Request Body
 	
(JSON)
	{
	"protocol":"*<protocol>*",
	"serverip":"*<serverip>*",
	"srcfile":"*<srcfile>*",
	"dstfile":"*<dstfile>*",
	"username":"*<username>*",
	"passwd":"*<passwd>*",
	"vrf_name":"*<vrf_name>*"
	}

where:

Element	Description
`protocol`	Protocol name; one of *tftp, sftp*.
`serverip`	Server IP address.
`srcfile`	Source file; up to 256 characters long.
`dstfile`	Destination file; one of *running_config, startup_config*.
`username`	(Optional) Username for the server.
`passwd`	(Optional) Password for the server username.
`vrf_name`	(Optional) VRF name; an alphabetic string up to 64 characters long.

Response

Response Body
 	
(JSON)
	{
	  "status": "*<status>*",
	  "details": "*<details>*",
	  "filename": "*<filename>*"
	}

where:

Element	Description
`status`	Transfer status; one of *transferring, installing, successful, failed*.
`details`	Detailed description of the status; one of: • *Transferring running-config* • *Transferring startup-config* • *Installing image* • *image installation succeeded* • *Copy success* • *VRF vrf_name doesn't exist* • *Another image installation is in progress* • *Host serverip is unreachable* • *ONIE feature in not enabled on this switch* • *File not found* • *SFTP authentication failure* • *image installation failed* • *Copy failed*
`filename`	Configuration filename.

Upload Tech Support to Server

Uploads technical support information from the switch to the server.

Request

Method Type	POST
Request URI	/nos/api/upload/tech_support

Request Body
 	
(JSON)
	{
	"protocol":"*<protocol>*",
	"serverip":"*<serverip>*",
	"dstfile":"*<dstfile>*",
	"username":"*<username>*",
	"passwd":"*<passwd>*",
	"vrf_name":"*<vrf_name>*"
	}

where:

Element	Description
`protocol`	Protocol name; one of *tftp, sftp*.
`serverip`	Server IP address.
`dstfile`	Destination file; one of *running_config, startup_config*.
`username`	(Optional) Username for the server.
`passwd`	(Optional) Password for the server username.
`vrf_name`	(Optional) VRF name; an alphabetic string up to 64 characters long.

Response

Response Body
 	
(JSON)
	{
	"status": "*<status>*",
	}

where:

Element	Description
`status`	Transfer status; one of *transferring, installing, successful, failed*.

Get Download Transfer Status

Gets the status of a downloading transfer.

Request

Method Type	GET
Request URI	/nos/api/download/status/<*content*>
Request Body (JSON)	

where:

Element	Description
content	One of `image`, `running_config`, `startup_config`.

Response

Response Body
 	
(JSON)
	`{`
	`  "status": "`<*status*>`",`
	`  "details": "`<*details*>`",`
	`  "filename": "`<*filename*>`"`
	`}`

where:

Element	Description
`status`	Transfer status; one of *transferring, installing, successful, failed*.
`details`	Detailed description of the status; one of: • *Transferring running-config* • *Transferring startup-config* • *Installing image* • *image installation succeeded* • *Copy success* • *VRF vrf_name doesn't exist* • *Another image installation is in progress* • *Host serverip is unreachable* • *ONIE feature in not enabled on this switch* • *File not found* • *SFTP authentication failure* • *image installation failed* • *Copy failed*
`filename`	Name of file being downloaded; up to 256 characters long.

Get Upload Transfer Status

Gets the status of an uploading transfer.

Request

Method Type	`GET`
Request URI	/nos/api/upload/status/<*content*>
Request Body (JSON)	

where:

Element	Description
content	One of `image`, `running_config`, `startup_config`.

Response

Response Body
 	
(JSON)
	`{`
	`  "status": "`<*status*>`",`
	`  "details": "`<*details*>`",`
	`  "filename": "`<*filename*>`"`
	`}`

where:

Element	Description
`status`	Transfer status; one of *transferring, installing, successful, failed*.

Element	Description
`details`	Detailed description of the status; one of: • *Transferring running-config* • *Transferring startup-config* • *Transferring tech-support* • *Copy success* • *VRF vrf_name doesn't exist* • *Another image installation is in progress* • *Host serverip is unreachable* • *ONIE feature in not enabled on this switch* • *File not found* • *SFTP authentication failure* • *Copy failed*
`filename`	Name of file being saved on the server; up to 256 characters long.

NPA

Note: Nutanix VDM is supported on the following Lenovo switches:

- RackSwitch G8272
- RackSwitch G8332
- RackSwitch G8296
- ThinkSystem NE1032T RackSwitch
- ThinkSystem NE1032 RackSwitch
- ThinkSystem NE1072T RackSwitch
- ThinkSystem NE10032 RackSwitch
- ThinkSystem NE2572 RackSwitch

The following Network Policy Agent (NPA) URIs are available:

- /nos/api/info/npa/vm/nutanix GET
- /nos/api/info/npa/vm/nutanix?uuid="*<uuid>*" GET
- /nos/api/info/npa/vm/nutanix?name="*<name>*" GET
- /nos/api/info/npa/vm/nutanix?if_name="*<if_name>*" GET
- /nos/api/info/npa/vnetwork/nutanix GET
- /nos/api/info/npa/vnetwork/nutanix?uuid="*<uuid>*" GET
- /nos/api/info/npa/stats/vnic/nutanix GET
- /nos/api/info/npa/stats/vnic/nutanix?uuid="*<vm_uuid>*" GET
- /nos/api/info/npa/stats/vnic/nutanix?name="*<vm_name>*" GET
- /nos/api/info/npa/stats/vnic/nutanix?if_name="*<if_name>*" GET
- /nos/api/info/npa/vm/vmware GET
- /nos/api/info/npa/vm/vmware?uuid="*<uuid>*" GET
- /nos/api/info/npa/vm/vmware?name="*<name>*" GET
- /nos/api/info/npa/vm/vmware?if_name="*<if_name>*" GET
- /nos/api/info/npa/vnetwork/vmware GET
- /nos/api/info/npa/stats/vnic/vmware GET
- /nos/api/info/npa/stats/vnic/vmware?uuid="*<vm_uuid>*" GET
- /nos/api/info/npa/stats/vnic/vmware?name="*<vm_name>*" GET
- /nos/api/info/npa/stats/vnic/vmware?if_name="*<if_name>*" GET

The following NPA commands are available (Nutanix):

- Get VM Information
- Get VM Information by VM UUID
- Get VM Information by VM Name

- Get VM Interface Information
- Get VM Information for Specific Interface
- Get Virtual Network Information
- Get Virtual Network Information by UUID
- Get VNIC Statistics
- Get VNIC Statistics for Specific VM by VM UUID
- Get VNIC Statistics for Specific VM by VM Name
- Get VNIC Interface Statistics
- Get VNIC Statistics for Specific Interface

The following NPA commands are available (VMware):

- Get All VM Information
- Get Specific VM Information by VM UUID
- Get Specific VM Information by VM Name
- Get VM Interface Information
- Get VM Information for Specific Interface
- Get Virtual Network Information
- Get VNIC Statistics
- Get VNIC Statistics for Specific VM by VM UUID
- Get VNIC Statistics for Specific VM by VM Name
- Get VNIC Interface Statistics
- Get VNIC Statistics for Specific Interface

Get VM Information

Gets information about configured virtual machines (VMs).

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/nutanix
Request Body (JSON)	

Response

Response Body (JSON)

```
[
  {
     nic_list:
        [
          {
            mac: <value>,
            uuid: <value>
          }
        ]
      uuid: <value>,
      num_cores_per_vcpu: <value>,
      hypervisor_type: <value>,
      memory_size_mib: <value>,
      num_vcpus: <value>,
      power_state: {on|off},
      host_uuid: <value>,
      name: <value>
  }
]
```

where:

Element	Description
`nic_list`	List of dictionary containing VNIC information.
`mac`	The MAC address of the VM. Valid value: MAC address in the following format: XX:XX:XX:XX:XX:XX.
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.
`num_cores_per_vcpu`	The number of cores for each virtual CPU (integer).
`hypervisor_type`	The type of hypervisor (string).
`memory_size_mib`	The size of the allocated memory in MBs (integer).
`num_vcpus`	The number of virtual CPUs (integer).

Element	Description
`power_ state`	The power state of the VM; one of *on, off*.
`host_uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`name`	The name of the VM (string).

Get VM Information by VM UUID

Gets information about a specific virtual machine (VM) by its Universal Unique Identifier (UUID).

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/nutanix?uuid=<*uuid*>
Request Body (JSON)	

where:

Element	Description
uuid	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.

Response

Response Body (JSON)

```
[
  {
     nic_list:
        [
          {
            mac: <value>,
            uuid: <value>
          }
        ]
      uuid: <value>,
      num_cores_per_vcpu: <value>,
      hypervisor_type: <value>,
      memory_size_mib: <value>,
      num_vcpus: <value>,
      power_state: {on|off},
      host_uuid: <value>,
      name: <value>
  }
]
```

where:

Element	Description
`nic_list`	List of dictionary containing VNIC information.
`mac`	The MAC address of the VM. Valid value: the MAC address in the following format: *XX:XX:XX:XX:XX:XX*.
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.

Element	Description
num_cores_per_vcpu	The number of cores for each virtual CPU (integer).
hypervisor_type	The type of hypervisor (string).
memory_size_mib	The size of the allocated memory in MBs (integer).
num_vcpus	The number of virtual CPUs (integer).
power_state	The power state of the VM; one of *on, off*.
host_uuid	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
name	The name of the VM (string).

Get VM Information by VM Name

Gets information about a specific virtual machine (VM) by its name.

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/nutanix?name=<*name*>
Request Body (JSON)	

where:

Element	Description
name	The name of the VM (string).

Response

Response Body (JSON):

```
[
  {
     nic_list:
        [
          {
            mac: <value>,
            uuid: <value>
          }
        ]
      uuid: <value>,
      num_cores_per_vcpu: <value>,
      hypervisor_type: <value>,
      memory_size_mib: <value>,
      num_vcpus: <value>,
      power_state: {on|off},
      host_uuid: <value>,
      name: <value>
  }
]
```

where:

Element	Description
`nic_list`	List of dictionary containing VNIC information.
`mac`	The MAC address of the VM. Valid value: the MAC address in the following format: *XX:XX:XX:XX:XX:XX*.
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.
`num_cores_per_vcpu`	The number of cores for each virtual CPU (integer).

Element	Description
`hypervisor_type`	The type of hypervisor (string).
`memory_size_mib`	The size of the allocated memory in MBs (integer).
`num_vcpus`	The number of virtual CPUs (integer).
`power_state`	The power state of the VM; one of *on, off*.
`host_uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`name`	The name of the VM (string).

Get VM Interface Information

Gets information about virtual machines (VMs) for all switch interfaces.

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/nutanix?if_name= or /nos/api/info/npa/vm/nutanix?if_name=all
Request Body (JSON)	

Response

Response Body (JSON)	{
	  if_name: <*value*>
	  [
	  {
	  nic_list:
	  [
	  {
	  mac: <*value*>,
	  uuid: <*value*>
	  }
	  ]
	  uuid: <*value*>,
	  num_cores_per_vcpu: <*value*>,
	  hypervisor_type: <*value*>,
	  memory_size_mib: <*value*>,
	  num_vcpus: <*value*>,
	  power_state: {on|off},
	  host_uuid: <*value*>,
	  name: <*value*>
	  }
	  ]
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`nic_list`	List of dictionary containing VNIC information.
`mac`	The MAC address of the VM. Valid value: the MAC address in the following format: XX:XX:XX:XX:XX:XX.
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.
`num_cores_per_vcpu`	The number of cores for each virtual CPU (integer).

Element	Description
`hypervisor_type`	The type of hypervisor (string).
`memory_size_mib`	The size of the allocated memory in MBs (integer).
`num_vcpus`	The number of virtual CPUs (integer).
`power_state`	The power state of the VM; one of *on, off*.
`host_uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`name`	The name of the VM (string).

Get VM Information for Specific Interface

Gets information about virtual machines (VMs) for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/nutanix?if_name=<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Response

Response Body (JSON)

```
{
  if_name: <value>
    [
      {
         nic_list:
            [
              {
                mac: <value>,
                uuid: <value>
              }
            ]
          uuid: <value>,
          num_cores_per_vcpu: <value>,
          hypervisor_type: <value>,
          memory_size_mib: <value>,
          num_vcpus: <value>,
          power_state: {on|off},
          host_uuid: <value>,
          name: <value>
      }
    ]
}
```

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`nic_list`	List of dictionary containing VNIC information.
`mac`	The MAC address of the VM. Valid value: the MAC address in the following format: *XX:XX:XX:XX:XX:XX*.

Element	Description
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.
`num_cores_per_vcpu`	The number of cores for each virtual CPU (integer).
`hypervisor_type`	The type of hypervisor (string).
`memory_size_mib`	The size of the allocated memory in MBs (integer).
`num_vcpus`	The number of virtual CPUs (integer).
`power_state`	The power state of the VM; one of *on, off.*
`host_uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`name`	The name of the VM (string).

Get Virtual Network Information

Gets information about all virtual networks.

Request

Method Type	GET
Request URI	/nos/api/info/npa/vnetwork/nutanix
Request Body (JSON)	

Response

Response Body (JSON)	`[`
	`  {`
	`  uuid: <value>,`
	`  default_gateway: <value>,`
	`  network_address: <value>,`
	`  prefix_length: <value>,`
	`  vlan_id: <value>,`
	`  name: <value>`
	`  }`
	`]`

where:

Element	Description
`uuid`	The Universal Unique Identifier (UUID) of the virtual network; a string up to 36 characters long.
`default_gateway`	The virtual network default gateway. Valid value: the IP address.
`network_address`	The virtual network IP address. Valid value: the IP address.
`prefix_length`	The virtual network prefix length; an integer from *0-32*.
`vlan_id`	The VLAN ID for the virtual network; an integer from *0-3999*.
`name`	The name of the virtual network (string).

Get Virtual Network Information by UUID

Gets information about a specific virtual networks by its network Universal Unique Identifier (UUID).

Request

Method Type	GET
Request URI	/nos/api/info/npa/vnetwork/nutanix?uuid=<*uuid*>
Request Body (JSON)	

where:

Element	Description
uuid	The Universal Unique Identifier (UUID) of the virtual network; a string up to 36 characters long.

Response

Response Body (JSON)	[
	  {
	  uuid: <*value*>,
	  default_gateway: <*value*>,
	  network_address: <*value*>,
	  prefix_length: <*value*>,
	  vlan_id: <*value*>,
	  name: <*value*>
	  }
	]

where:

Element	Description
`uuid`	The Universal Unique Identifier (UUID) of the virtual network; a string up to 36 characters long.
`default_gateway`	The virtual network default gateway. Valid value: the IP address.
`network_address`	The virtual network IP address.
`prefix_length`	The virtual network prefix length; an integer from *0-32.*
`vlan_id`	The VLAN ID for the virtual network; an integer from *0-3999.*
`name`	The name of the virtual network (string).

Get VNIC Statistics

Gets all virtualized Network Interface Card (vNIC) statistics.

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/nutanix
Request Body (JSON)	

Response

Response Body (JSON)

```
[
  {
    vm_uuid: <value>,
    vnic_uuid: <value>,
    time_stamp: <value>,
    bytes_rx: <value>,
    bytes_tx: <value>,
    pkts_rx: <value>,
    pkts_tx: <value>,
    mcast_pkts_rx: <value>,
    mcast_pkts_tx: <value>,
    bcast_pkts_rx: <value>,
    bcast_pkts_tx: <value>,
    dropped_pkts_rx: <value>,
    dropped_pkts_tx: <value>,
    error_pkts_rx: <value>,
    error_pkts_tx: <value>,
    rate_kbps_rx: <value>,
    rate_kbps_tx: <value>,
    unsupported_prot_pkts_rx: <value>,
    rate_usage_kbps: <value>
  }
]
```

where:

Element	Description
`vm_uuid`	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
`vnic_uuid`	The UUID of the vNIC; a string up to 36 characters long.
`time_stamp`	The time when the vNIC statistics were collected. Valid value: the *timestamp* in the following format: *YYYY/MM/DD HH:MM:SS*.
`bytes_rx`	The number of received bytes (integer).
`bytes_tx`	The number of transmitted bytes (integer).
`pkts_rx`	The number of received packets (integer).

Element	Description
pkts_tx	The number of transmitted packets (integer).
mcast_pkts_rx	The number of received multicast packets (integer).
mcast_pkts_tx	The number of transmitted multicast packets (integer).
bcast_pkts_rx	The number of received broadcast packets (integer).
bcast_pkts_tx	The number of transmitted broadcast packets (integer).
dropped_pkts_rx	The number of received dropped packets (integer).
dropped_pkts_tx	The number of transmitted dropped packets (integer).
error_pkts_rx	The number of received error packets (integer).
error_pkts_tx	The number of transmitted error packets (integer).
rate_kbps_rx	The rate of received data, in kbps (integer).
rate_kbps_rx	The rate of transmitted data, in kbps (integer).
unsupported_prot_pkts_rx	The number of received unsupported protocol packets (integer).
rate_usage_kbps	The usage rate, in kbps (integer).

Get VNIC Statistics for Specific VM by VM UUID

Gets virtualized Network Interface Card (vNIC) statistics for a specific virtual machine (VM) by its Universal Unique Identifier (UUID).

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/nutanix?uuid=<*vm_uuid*>
Request Body (JSON)	

where:

Element	Description
vm_uuid	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.

Response

Response Body (JSON)

```
[
  {
    vm_uuid: <value>,
    vnic_uuid: <value>,
    time_stamp: <value>,
    bytes_rx: <value>,
    bytes_tx: <value>,
    pkts_rx: <value>,
    pkts_tx: <value>,
    mcast_pkts_rx: <value>,
    mcast_pkts_tx: <value>,
    bcast_pkts_rx: <value>,
    bcast_pkts_tx: <value>,
    dropped_pkts_rx: <value>,
    dropped_pkts_tx: <value>,
    error_pkts_rx: <value>,
    error_pkts_tx: <value>,
    rate_kbps_rx: <value>,
    rate_kbps_tx: <value>,
    unsupported_prot_pkts_rx: <value>,
    rate_usage_kbps: <value>
  }
]
```

where:

Element	Description
vm_uuid	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
vnic_uuid	The UUID of the vNIC; a string up to 36 characters long.
time_stamp	The time when the vNIC statistics were collected. Valid value: the timestamp in the following format: *YYYY/MM/DD HH:MM:SS*.
bytes_rx	The number of received bytes (integer).
bytes_tx	The number of transmitted bytes (integer).
pkts_rx	The number of received packets (integer).
pkts_tx	The number of transmitted packets (integer).
mcast_pkts_rx	The number of received multicast packets (integer).
mcast_pkts_tx	The number of transmitted multicast packets (integer).
bcast_pkts_rx	The number of received broadcast packets (integer).
bcast_pkts_tx	The number of transmitted broadcast packets (integer).
dropped_pkts_rx	The number of received dropped packets (integer).
dropped_pkts_tx	The number of transmitted dropped packets (integer).
error_pkts_rx	The number of received error packets (integer).
error_pkts_tx	The number of transmitted error packets (integer).
rate_kbps_rx	The rate of received data, in kbps (integer).
rate_kbps_rx	The rate of transmitted data, in kbps (integer).
unsupported_prot_pkts_rx	The number of received unsupported protocol packets (integer).
rate_usage_kbps	The usage rate, in kbps (integer).

Get VNIC Statistics for Specific VM by VM Name

Gets virtualized Network Interface Card (vNIC) statistics for a specific virtual machine (VM) by its name.

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/nutanix?name=<*vm_name*>
Request Body (JSON)	

where:

Element	Description
`name`	The name of the VM (string).

Response

Response Body (JSON)

```
[
  {
    vm_uuid: <value>,
    vnic_uuid: <value>,
    time_stamp: <value>,
    bytes_rx: <value>,
    bytes_tx: <value>,
    pkts_rx: <value>,
    pkts_tx: <value>,
    mcast_pkts_rx: <value>,
    mcast_pkts_tx: <value>,
    bcast_pkts_rx: <value>,
    bcast_pkts_tx: <value>,
    dropped_pkts_rx: <value>,
    dropped_pkts_tx: <value>,
    error_pkts_rx: <value>,
    error_pkts_tx: <value>,
    rate_kbps_rx: <value>,
    rate_kbps_tx: <value>,
    unsupported_prot_pkts_rx: <value>,
    rate_usage_kbps: <value>
  }
]
```

where:

Element	Description
vm_uuid	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
vnic_uuid	The UUID of the vNIC; a string up to 36 characters long.
time_stamp	The time when the vNIC statistics were collected. Valid value: the *timestamp* in the following format: *YYYY/MM/DD HH:MM:SS*.
bytes_rx	The number of received bytes (integer).
bytes_tx	The number of transmitted bytes (integer).
pkts_rx	The number of received packets (integer).
pkts_tx	The number of transmitted packets (integer).
mcast_pkts_rx	The number of received multicast packets (integer).
mcast_pkts_tx	The number of transmitted multicast packets (integer).
bcast_pkts_rx	The number of received broadcast packets (integer).
bcast_pkts_tx	The number of transmitted broadcast packets (integer).
dropped_pkts_rx	The number of received dropped packets (integer).
dropped_pkts_tx	The number of transmitted dropped packets (integer).
error_pkts_rx	The number of received error packets (integer).
error_pkts_tx	The number of transmitted error packets (integer).
rate_kbps_rx	The rate of received data, in kbps (integer).
rate_kbps_rx	The rate of transmitted data, in kbps (integer).
unsupported_prot_pkts_rx	The number of received unsupported protocol packets (integer).
rate_usage_kbps	The usage rate, in kbps (integer).

Get VNIC Interface Statistics

Gets virtualized Network Interface Card (vNIC) statistics for all switch interfaces.

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/nutanix?if_name= or /nos/api/info/npa/stats/vnic/nutanix?if_name=all
Request Body (JSON)	

Response

Response Body (JSON)

```
{
  if_name: <value>,
    [
      {
        vm_uuid: <value>,
        vnic_uuid: <value>,
        time_stamp: <value>,
        bytes_rx: <value>,
        bytes_tx: <value>,
        pkts_rx: <value>,
        pkts_tx: <value>,
        mcast_pkts_rx: <value>,
        mcast_pkts_tx: <value>,
        bcast_pkts_rx: <value>,
        bcast_pkts_tx: <value>,
        dropped_pkts_rx: <value>,
        dropped_pkts_tx: <value>,
        error_pkts_rx: <value>,
        error_pkts_tx: <value>,
        rate_kbps_rx: <value>,
        rate_kbps_tx: <value>,
        unsupported_prot_pkts_rx: <value>,
        rate_usage_kbps: <value>
      }
    ]
}
```

where:

Element	Description
`if_name`	The name of the switch interface. For example *Ethernet1/12*.
`vm_uuid`	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
`vnic_uuid`	The UUID of the vNIC; a string up to 36 characters long.
`time_stamp`	The time when the vNIC statistics were collected. Valid value: the *timestamp* in the following format: *YYYY/MM/DD HH:MM:SS*.
`bytes_rx`	The number of received bytes (integer).
`bytes_tx`	The number of transmitted bytes (integer).
`pkts_rx`	The number of received packets (integer).
`pkts_tx`	The number of transmitted packets (integer).
`mcast_pkts_rx`	The number of received multicast packets (integer).
`mcast_pkts_tx`	The number of transmitted multicast packets (integer).
`bcast_pkts_rx`	The number of received broadcast packets (integer).
`bcast_pkts_tx`	The number of transmitted broadcast packets (integer).
`dropped_pkts_rx`	The number of received dropped packets (integer).
`dropped_pkts_tx`	The number of transmitted dropped packets (integer).
`error_pkts_rx`	The number of received error packets (integer).
`error_pkts_tx`	The number of transmitted error packets (integer).
`rate_kbps_rx`	The rate of received data, in kbps (integer).
`rate_kbps_rx`	The rate of transmitted data, in kbps (integer).
`unsupported_prot_pkts_rx`	The number of received unsupported protocol packets (integer).
`rate_usage_kbps`	The usage rate, in kbps (integer).

Get VNIC Statistics for Specific Interface

Gets virtualized Network Interface Card (vNIC) statistics for a specific switch interfaces.

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/nutanix?if_name=<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Response

Response Body (JSON)

```
{
  if_name: <value>,
    [
      {
        vm_uuid: <value>,
        vnic_uuid: <value>,
        time_stamp: <value>,
        bytes_rx: <value>,
        bytes_tx: <value>,
        pkts_rx: <value>,
        pkts_tx: <value>,
        mcast_pkts_rx: <value>,
        mcast_pkts_tx: <value>,
        bcast_pkts_rx: <value>,
        bcast_pkts_tx: <value>,
        dropped_pkts_rx: <value>,
        dropped_pkts_tx: <value>,
        error_pkts_rx: <value>,
        error_pkts_tx: <value>,
        rate_kbps_rx: <value>,
        rate_kbps_tx: <value>,
        unsupported_prot_pkts_rx: <value>,
        rate_usage_kbps: <value>
      }
    ]
}
```

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`vm_uuid`	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
`vnic_uuid`	The UUID of the vNIC; a string up to 36 characters long.
`time_stamp`	The time when the vNIC statistics were collected. Valid value: the *timestamp* in the following format: *YYYY/MM/DD HH:MM:SS*.
`bytes_rx`	The number of received bytes (integer).
`bytes_tx`	The number of transmitted bytes (integer).
`pkts_rx`	The number of received packets (integer).
`pkts_tx`	The number of transmitted packets (integer).
`mcast_pkts_rx`	The number of received multicast packets (integer).
`mcast_pkts_tx`	The number of transmitted multicast packets (integer).
`bcast_pkts_rx`	The number of received broadcast packets (integer).
`bcast_pkts_tx`	The number of transmitted broadcast packets (integer).
`dropped_pkts_rx`	The number of received dropped packets (integer).
`dropped_pkts_tx`	The number of transmitted dropped packets (integer).
`error_pkts_rx`	The number of received error packets (integer).
`error_pkts_tx`	The number of transmitted error packets (integer).
`rate_kbps_rx`	The rate of received data, in kbps (integer).
`rate_kbps_rx`	The rate of transmitted data, in kbps (integer).
`unsupported_prot_pkts_rx`	The number of received unsupported protocol packets (integer).
`rate_usage_kbps`	The usage rate, in kbps (integer).

Get All VM Information

Gets information about configured virtual machines (VMs).

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/vmware
Request Body (JSON)	

Response

Response Body (JSON)	[{
	  nic_list: [{
	  connected_state: {connected|disconnected},
	  nic_type: <*value*>,
	  mac: <*value*>,
	  network_name: <*value*>
	  }],
	  uuid: <*value*>,
	  num_cores_per_vcpu: <*value*>,
	  memory_size_mb: <*value*>,
	  num_vcpus: <*value*>,
	  power_state: <*poweredOff|poweredOn*>,
	  host_uuid: <*value*>,
	  name: <*value*>
	}

where:

Element	Description
nic_list	List of dictionary containing VNIC information.
connected_state	Whether the VNIC is connected or disconnected (string).
nic_type	The type of VMware VNIC (string).
mac	The MAC address of the VM. Valid value: the MAC address in the following format: XX:XX:XX:XX:XX:XX.
network_name	The name of the network (string).
uuid	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.
num_cores_per_vcpu	The number of cores for each virtual CPU (integer).
memory-size	The size of the allocated memory in MBs (integer).
num_vcpus	The number of virtual CPUs, an integer.
power_state	The power state of the VM; one of *poweredOff, poweredOn*.

Element	Description
`host_uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`name`	The name of the VM (string).

Get Specific VM Information by VM UUID

Gets information about a specific virtual machine (VM) by its Universal Unique Identifier (UUID).

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/vmware?uuid=<*uuid*>
Request Body (JSON)	

where:

Element	Description
vm_uuid	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.

Response

Response Body (JSON)	[{
	  nic_list: [{
	  connected_state: {connected|disconnected},
	  nic_type: <*value*>,
	  mac: <*value*>,
	  network_name: <*value*>
	  }],
	  uuid: <*value*>,
	  num_cores_per_vcpu: <*value*>,
	  memory_size_mb: <*value*>,
	  num_vcpus: <*value*>,
	  power_state: <*poweredOff|poweredOn*>,
	  host_uuid: <*value*>,
	  name: <*value*>
	}

where:

Element	Description
`nic_list`	List of dictionary containing VNIC information.
`connected_state`	Whether the VNIC is connected or disconnected (string).
`nic_type`	The type of VMware VNIC (string).
`mac`	The MAC address of the VM. Valid value: the MAC address in the following format: XX:XX:XX:XX:XX:XX.
`network_name`	The name of the network (string).
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.

Element	Description
num_cores_per_vcpu	The number of cores for each virtual CPU (integer).
memory-size	The size of the allocated memory in MBs (integer).
num_vcpus	The number of virtual CPUs, an integer.
power_state	The power state of the VM; one of *poweredOff, poweredOn*.
host_uuid	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
name	The name of the VM (string).

Get Specific VM Information by VM Name

Gets information about a specific virtual machine (VM) by its name.

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/vmware?name=<*name*>
Request Body (JSON)	

where:

Element	Description
vm_name	The name of the VM; a string.

Response

Response Body (JSON):

```
[{
  nic_list: [{
    connected_state: {connected|disconnected},
    nic_type: <value>,
    mac: <value>,
    network_name: <value>
  }],
  uuid: <value>,
  num_cores_per_vcpu: <value>,
  memory_size_mb: <value>,
  num_vcpus: <value>,
  power_state: <poweredOff|poweredOn>,
  host_uuid: <value>,
  name: <value>
}
```

where:

Element	Description
`nic_list`	List of dictionary containing VNIC information.
`connected_state`	Whether the VNIC is connected or disconnected (string).
`nic_type`	The type of VMware VNIC (string).
`mac`	The MAC address of the VM. Valid value: the MAC address in the following format: XX:XX:XX:XX:XX:XX.
`network_name`	The name of the network (string).
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.

Element	Description
`num_cores_per_vcpu`	The number of cores for each virtual CPU (integer).
`memory_size_mib`	The size of the allocated memory in MBs (integer).
`num_vcpus`	The number of virtual CPUs (integer).
`power_state`	The power state of the VM; one of *poweredOff, poweredOn*.
`host_uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`name`	The name of the VM (string).

Get VM Interface Information

Gets information about virtual machines (VMs) for all switch interfaces.

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/vmware?if_name= or /nos/api/info/npa/vm/vmware?if_name=all
Request Body (JSON)	

Response

Response Body (JSON)	{
	  if_name: [{
	  nic_list: [{
	  connected_state: <connected|disconnect>,
	  nic_type: *<value>*,
	  mac: *<value>*,
	  network_name: *<value>*
	  }],
	  uuid: *<value>*,
	  num_cores_per_vcpu: *<value>*,
	  memory_size_mb: *<value>*,
	  num_vcpus: *<value>*,
	  power_state: *<poweredOn|poweredOff>*,
	  host_uuid: *<value>*,
	  name: *<value>*
	  }]

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`nic_list`	List of dictionary containing VNIC information.
`connected_state`	Whether the VNIC is connected or disconnected (string).
`nic_type`	The type of VMware VNIC (string).
`mac`	The MAC address of the VM. Valid value: the MAC address in the following format: XX:XX:XX:XX:XX:XX.
`network_name`	The name of the network (string).
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.
`num_cores_per_vcpu`	The number of cores for each virtual CPU (integer).

Element	Description
`memory_ size_mib`	The size of the allocated memory in MBs (integer).
`num_vcpus`	The number of virtual CPUs (integer).
`power_ state`	The power state of the VM; one of *poweredOff, poweredOn.*
`host_uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`name`	The name of the VM (string).

Get VM Information for Specific Interface

Gets information about virtual machines (VMs) for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/npa/vm/vmware?if_name=<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Response

Response Body (JSON)	{
	  if_name: [{
	  nic_list: [{
	  connected_state: <connected|disconnect>,
	  nic_type: <*value*>,
	  mac: <*value*>,
	  network_name: <*value*>
	  }],
	  uuid: <*value*>,
	  num_cores_per_vcpu: <*value*>,
	  memory_size_mb: <*value*>,
	  num_vcpus: <*value*>,
	  power_state: <*poweredOn|poweredOff*>,
	  host_uuid: <*value*>,
	  name: <*value*>
	  }]

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`nic_list`	List of dictionary containing VNIC information.
`connected_state`	Whether the VNIC is connected or disconnected (string).
`nic_type`	The type of VMware VNIC (string).
`mac`	The MAC address of the VM. Valid value: the MAC address in the following format: XX:XX:XX:XX:XX:XX.
`network_name`	The name of the network (string).

Element	Description
`uuid`	The Universal Unique Identifier (UUID) of the VM; a string up to 36 characters long.
`num_cores_per_vcpu`	The number of cores for each virtual CPU (integer).
`memory_size_mib`	The size of the allocated memory in MBs (integer).
`num_vcpus`	The number of virtual CPUs (integer).
`power_state`	The power state of the VM; one of *poweredOff, poweredOn*.
`host_uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`name`	The name of the VM (string).

Get Virtual Network Information

Gets information about all virtual networks.

Request

Method Type	GET
Request URI	/nos/api/info/npa/vnetwork/vmware
Request Body (JSON)	

Response

Response Body (JSON)	`[`
	`  {`
	`  vswitch_name: <value>,`
	`  uuid: <value>,`
	`  vlan_id: <value>,`
	`  name: <value>`
	`  }`
	`]`

where:

Element	Description
`vswitch_name`	The name of the Vswitch (string).
`uuid`	The Universal Unique Identifier (UUID) of the host; a string up to 36 characters long.
`vlan_id`	The VLAN ID for the virtual network; an integer from 0-3999.
`name`	The name of the virtual network (string).

Get VNIC Statistics

Gets all virtualized Network Interface Card (vNIC) statistics.

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/vmware
Request Body (JSON)	

Response

Response Body (JSON)	[{
	pkts_tx: *<value>*,
	bcast_pkts_rx: *<value>*,
	dropped_pkts_tx: *<value>*,
	bcast_pkts_tx: *<value>*,
	pkts_rx: *<value>*,
	bytes_rx: *<value>*,
	rate_usage_kbps: *<value>*,
	mcast_pkts_tx: *<value>*,
	rate_kbps_tx: *<value>*,
	rate_kbps_rx: *<value>*,
	vm_uuid: *<value>*,
	dropped_pkts_rx: *<value>*,
	bytes_tx: *<value>*,
	time_stamp: *<value>*,
	vnic_mac: *<value>*,
	mcast_pkts_rx: *<value>*
	}

where:

Element	Description
vm_uuid	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
vnic_mac	The vNIC MAC address in the following format: xx:xx:xx:xx:xx:xx.
time_stamp	The time when the vNIC statistics were collected in the following format: YYYY/MM/DD HH:MM:SS.
bytes_rx	The number of received bytes (integer).
bytes_tx	The number of transmitted bytes (integer).
pkts_rx	The number of received packets (integer).
pkts_tx	The number of transmitted packets (integer).
mcast_pkts_rx	The number of received multicast packets (integer).

Element	Description
mcast_pkts_tx	The number of transmitted multicast packets (integer).
bcast_pkts_rx	The number of received broadcast packets (integer).
bcast_pkts_tx	The number of transmitted broadcast packets (integer).
dropped_pkts_rx	The number of received dropped packets (integer).
dropped_pkts_tx	The number of transmitted dropped packets (integer).
rate_kbps_rx	The rate of received data, in kbps (integer).
rate_kbps_rx	The rate of transmitted data, in kbps (integer).
rate_usage_kbps	The usage rate, in kbps (integer).

Get VNIC Statistics for Specific VM by VM UUID

Gets virtualized Network Interface Card (vNIC) statistics for a specific virtual machine (VM) by its Universal Unique Identifier (UUID).

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/vmware?uuid=<*vm_uuid*>
Request Body (JSON)	

where:

Element	Description
vm_uuid	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.

Response

Response Body (JSON)	[{
	pkts_tx: <*value*>,
	bcast_pkts_rx: <*value*>,
	dropped_pkts_tx: <*value*>,
	bcast_pkts_tx: <*value*>,
	pkts_rx: <*value*>,
	bytes_rx: <*value*>,
	rate_usage_kbps: <*value*>,
	mcast_pkts_tx: <*value*>,
	rate_kbps_tx: <*value*>,
	rate_kbps_rx: <*value*>,
	vm_uuid: <*value*>,
	dropped_pkts_rx: <*value*>,
	bytes_tx: <*value*>,
	time_stamp: <*value*>,
	vnic_mac: <*value*>,
	mcast_pkts_rx: <*value*>
	}

where:

Element	Description
`vm_uuid`	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
`vnic_mac`	The vNIC MAC address in the following format: `xx:xx:xx:xx:xx:xx.`
`time_stamp`	The time when the vNIC statistics were collected. Valid value: the *timestamp* in the following format: *YYYY/MM/DD HH:MM:SS*.
`bytes_rx`	The number of received bytes (integer).
`bytes_tx`	The number of transmitted bytes (integer).
`pkts_rx`	The number of received packets (integer).
`pkts_tx`	The number of transmitted packets (integer).
`mcast_pkts_rx`	The number of received multicast packets (integer).
`mcast_pkts_tx`	The number of transmitted multicast packets (integer).
`bcast_pkts_rx`	The number of received broadcast packets (integer).
`bcast_pkts_tx`	The number of transmitted broadcast packets (integer).
`dropped_pkts_rx`	The number of received dropped packets (integer).
`dropped_pkts_tx`	The number of transmitted dropped packets (integer).
`rate_kbps_rx`	The rate of received data, in kbps (integer).
`rate_kbps_rx`	The rate of transmitted data, in kbps (integer).
`rate_usage_kbps`	The usage rate, in kbps (integer).

Get VNIC Statistics for Specific VM by VM Name

Gets virtualized Network Interface Card (vNIC) statistics for a specific virtual machine (VM) by its name.

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/vmware?name=<*vm_name*>
Request Body (JSON)	

where:

Element	Description
vm_name	The name of the VM (string).

Response

Response Body (JSON)	[{
	pkts_tx: <*value*>,
	bcast_pkts_rx: <*value*>,
	dropped_pkts_tx: <*value*>,
	bcast_pkts_tx: <*value*>,
	pkts_rx: <*value*>,
	bytes_rx: <*value*>,
	rate_usage_kbps: <*value*>,
	mcast_pkts_tx: <*value*>,
	rate_kbps_tx: <*value*>,
	rate_kbps_rx: <*value*>,
	vm_uuid: <*value*>,
	dropped_pkts_rx: <*value*>,
	bytes_tx: <*value*>,
	time_stamp: <*value*>,
	vnic_mac: <*value*>,
	mcast_pkts_rx: <*value*>
	}

where:

Element	Description
`vm_uuid`	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
`vnic_mac`	The vNIC MAC address in the following format: `xx:xx:xx:xx:xx:xx.`
`time_stamp`	The time when the vNIC statistics were collected. Valid value: the *timestamp* in the following format: *YYYY/MM/DD HH:MM:SS*.
`bytes_rx`	The number of received bytes (integer).
`bytes_tx`	The number of transmitted bytes (integer).
`pkts_rx`	The number of received packets (integer).
`pkts_tx`	The number of transmitted packets (integer).
`mcast_pkts_rx`	The number of received multicast packets (integer).
`mcast_pkts_tx`	The number of transmitted multicast packets (integer).
`bcast_pkts_rx`	The number of received broadcast packets (integer).
`bcast_pkts_tx`	The number of transmitted broadcast packets (integer).
`dropped_pkts_rx`	The number of received dropped packets (integer).
`dropped_pkts_tx`	The number of transmitted dropped packets (integer).
`rate_kbps_rx`	The rate of received data, in kbps (integer).
`rate_kbps_rx`	The rate of transmitted data, in kbps (integer).
`rate_usage_kbps`	The usage rate, in kbps (integer).

Get VNIC Interface Statistics

Gets virtualized Network Interface Card (vNIC) statistics for all switch interfaces.

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/vmware?if_name= or /nos/api/info/npa/stats/vnic/vmware?if_name=all
Request Body (JSON)	

Response

Response Body (JSON)	if_name :[{
	pkts_tx: <*value*>,
	bcast_pkts_rx: <*value*>,
	dropped_pkts_tx: <*value*>,
	bcast_pkts_tx: <*value*>,
	pkts_rx: <*value*>,
	bytes_rx: <*value*>,
	rate_usage_kbps: <*value*>,
	mcast_pkts_tx: <*value*>,
	rate_kbps_tx: <*value*>,
	rate_kbps_rx: <*value*>,
	vm_uuid: <*value*>,
	dropped_pkts_rx: <*value*>,
	bytes_tx: <*value*>,
	time_stamp: <*value*>,
	vnic_mac: <*value*>,
	mcast_pkts_rx: <*value*>
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`vm_uuid`	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
`vnic_mac`	The vNIC MAC address in the following format: `xx:xx:xx:xx:xx:xx.`
`time_stamp`	The time when the vNIC statistics were collected. Valid value: the *timestamp* in the following format: *YYYY/MM/DD HH:MM:SS*.
`bytes_rx`	The number of received bytes (integer).
`bytes_tx`	The number of transmitted bytes (integer).
`pkts_rx`	The number of received packets (integer).
`pkts_tx`	The number of transmitted packets (integer).
`mcast_pkts_rx`	The number of received multicast packets (integer).
`mcast_pkts_tx`	The number of transmitted multicast packets (integer).
`bcast_pkts_rx`	The number of received broadcast packets (integer).
`bcast_pkts_tx`	The number of transmitted broadcast packets (integer).
`dropped_pkts_rx`	The number of received dropped packets (integer).
`dropped_pkts_tx`	The number of transmitted dropped packets (integer).
`rate_kbps_rx`	The rate of received data, in kbps (integer).
`rate_kbps_rx`	The rate of transmitted data, in kbps (integer).
`rate_usage_kbps`	The usage rate, in kbps (integer).

Get VNIC Statistics for Specific Interface

Gets virtualized Network Interface Card (vNIC) statistics for a specific switch interfaces.

Request

Method Type	GET
Request URI	/nos/api/info/npa/stats/vnic/vmware?if_name=<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Response

Response Body (JSON)	`if_name :[{`
	`  pkts_tx: <value>,`
	`  bcast_pkts_rx: <value>,`
	`  dropped_pkts_tx: <value>,`
	`  bcast_pkts_tx: <value>,`
	`  pkts_rx: <value>,`
	`  bytes_rx: <value>,`
	`  rate_usage_kbps: <value>,`
	`  mcast_pkts_tx: <value>,`
	`  rate_kbps_tx: <value>,`
	`  rate_kbps_rx: <value>,`
	`  vm_uuid: <value>,`
	`  dropped_pkts_rx: <value>,`
	`  bytes_tx: <value>,`
	`  time_stamp: <value>,`
	`  vnic_mac: <value>,`
	`  mcast_pkts_rx: <value>`
	`}`

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`vm_uuid`	The Universal Unique Identifier (UUID) of the virtual machine (VM); a string up to 36 characters long.
`vnic_mac`	The vNIC MAC address in the following format: `xx:xx:xx:xx:xx:xx.`
`time_stamp`	The time when the vNIC statistics were collected. Valid value: the *timestamp* in the following format: *YYYY/MM/DD HH:MM:SS*.
`bytes_rx`	The number of received bytes (integer).
`bytes_tx`	The number of transmitted bytes (integer).
`pkts_rx`	The number of received packets (integer).
`pkts_tx`	The number of transmitted packets (integer).
`mcast_pkts_rx`	The number of received multicast packets (integer).
`mcast_pkts_tx`	The number of transmitted multicast packets (integer).
`bcast_pkts_rx`	The number of received broadcast packets (integer).
`bcast_pkts_tx`	The number of transmitted broadcast packets (integer).
`dropped_pkts_rx`	The number of received dropped packets (integer).
`dropped_pkts_tx`	The number of transmitted dropped packets (integer).
`rate_kbps_rx`	The rate of received data, in kbps (integer).
`rate_kbps_rx`	The rate of transmitted data, in kbps (integer).
`rate_usage_kbps`	The usage rate, in kbps (integer).

NTP

The following NTP URIs are available:

- /nos/api/cfg/ntp/peers GET, POST, DELETE
- /nos/api/cfg/ntp/authentication-keys GET, POST, DELETE

The following NTP commands are available:

- Get NTP Properties
- Update NTP Servers and Peers
- Delete NTP Servers and Peers
- Get NTP Authentication Keys
- Set NTP Authentication Keys
- Delete NTP Authentication Keys

Get NTP Properties

Gets the configured NTP servers and peers.

Note: This is required for XClarity support.

Request

Method Type	GET
Request URI	/nos/api/cfg/ntp/peers
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	{
	"type" : "*<server/peer>*"
	"name": *"<ip_address>"*,
	"server_type": *"<static/dynamic>"*,
	}
	]

where:

Element	Description
`type`	Configured server or peer.
`name`	IP address of peer/server.
`server_type`	Static or dynamic (server or peer).

Update NTP Servers and Peers

Updates the configured NTP servers and peers.

Note: This is required for XClarity support.

Request

Method Type	POST
Request URI	/nos/api/cfg/ntp/peers
Request Body (JSON)	{"type" : "<server | peer>", "name" : "<ip>", ["prefer" : "<yes/no>","minpoll" : "<value>", "maxpoll" : "<value>"] }

where:

Element	Description
type	Configured server or peer.
name	IP address of peer/server.
prefer	(Optional) One of: *Yes, No*.
minpoll	(Optional) Minimum poll value.
maxpoll	(Optional) Maximum poll value.

Response

Response Body
 	
(JSON)
	[
	{
	"type" : *"<server/peer>"*
	"name": *"<ip_address>"*,
	"prefer" : *"<yes/no>"*,
	"minpoll" : *"<value">*,
	"maxpoll" : *"<value>"*]
	}
	]

where:

Element	Description
type	Configured server or peer.
name	IP address of peer/server.
prefer	(Optional) One of: *Yes, No*.
minpoll	(Optional) Minimum poll value.
maxpoll	(Optional) Maximum poll value.

Delete NTP Servers and Peers

Deletes the configured NTP servers and peers.

Note: This is required for XClarity support.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/ntp/peers/<*ip*>/<*server or peer*>
Request Body (JSON)	

where:

Element	Description
<*ip*>/<*server or peer*>	IP address of peer/server.

Response

Response Body (JSON)	

Get NTP Authentication Keys

Gets NTP authentication keys.

Request

Method Type	GET
Request URI	/nos/api/cfg/ntp/authentication-keys
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  “key_num”: “<number>”,`
	`  “md5”: “<string>”  or`
	`  “sha1”:“<string>”`
	`  }`
	`]`

where:

Element	Description
`key`	The number of the trusted key (integer).
`md5, sha1`	(Mandatory) The MD5 or SHA1 authentication string.

Set NTP Authentication Keys

Sets NTP authentication keys.

Request

Method Type	POST
Request URI	/nos/api/cfg/ntp/authentication-keys

Request Body
 	
(JSON)
	`{ “key_num” : “<number>” ,`
	`“md5” : “<md5_string>”or`
	`“sha1”:“<sha1_string>”`
	`}`

where:

Element	Description
`key`	The number of the trusted key (integer).

Response

Response Body
 	
(JSON)
	`[`
	`{`
	`“key_num”: “<number>”,`
	`“md5”: “<string> ”or`
	`“sha1”:“<string>”`
	`}`
	`]`

where:

Element	Description
`key`	The number of the trusted key (integer).
`md5, sha1`	(Mandatory) The MD5 or SHA1 authentication string.

Delete NTP Authentication Keys

Deletes the specified NTP authentication keys.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/ntp/authentication-keys/<*key_number*>
Request Body (JSON)	

where:

Element	Description
key_number	The number of the trusted key (integer).

Response

`True` if the operation succeeded; otherwise `False`.

OSPF

The following OSPF URIs are available:

- /nos/api/info/ospf/stats GET
- /nos/api/info/ospf/traffic-stats GET
- /nos/api/info/ospf/neighbor GET
- /nos/api/info/ospf/route GET
- /nos/api/info/ospf/database GET
- /nos/api/info/ospf/border-router GET
- /nos/api/info/ospf/summary-address GET
- /nos/api/cfg/ospf/interface GET, PUT
- /nos/api/cfg/ospf/virtual-link GET, PUT
- /nos/api/info/ospf/process GET
- /nos/api/info/ospf/multiarea-neighbors GET
- /nos/api/info/ospf/ribcounter_info GET
- /nos/api/cfg/ospf/process PUT
- /nos/api/cfg/ospf/redistribute GET, PUT
- /nos/api/cfg/ospf/nssa GET, PUT
- /nos/api/cfg/ospf/area_def_cost PUT
- /nos/api/cfg/ospf/area_auth PUT
- /nos/api/cfg/ospf/summary_addr PUT
- /nos/api/cfg/ospf/range PUT
- /nos/api/cfg/ospf/overflow_db PUT
- /nos/api/cfg/ospf/autocost_refbw PUT
- /nos/api/cfg/ospf/stub PUT
- /nos/api/cfg/ospf/clear PUT

The following OSPF interface property commands are available:

- Get OSPF Global Statistics
- Get OSPF Traffic Statistics
- Get OSPF Neighbors
- Get OSPF Routes
- Get OSPF Database
- Get OSPF Border Routers
- Get OSPF Summary Address

- Get OSPF Interface
- Set OSPF Interface
- Get OSPF Virtual Links
- Set OSPF Virtual Links
- Get OSPF Process
- Get OSPF Multi-Area Neighbor
- Get OSPF RIB Counters
- Set OSPF Process
- Get OSPF Redistribute
- Set OSPF Redistribute
- Get OSPF NSSA area
- Set OSPF NSSA area
- Set OSPF default cost
- Set Area Authentication
- Set OSPF Summary Address
- Set OSPF Area Range
- Set OSPF Overflow Database
- Set OSPF Auto-cost Reference Bandwidth
- Set OSPF Stub Configuration
- Set OSPF Remove Configuration

Get OSPF Global Statistics

Gets the global OSPF statistics.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/stats/*<vrf_name>*
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body (JSON)	[{ "stats": { "ospf_id": "*<ospf_id>*" "clr_timer_str": "{ <*clr_timer_str*>", "router_id_changes": "<*router_id_changes*>", "dr_election_counter": "<*dr_election_counter*>", "older_lsas_counter": "<*older_lsas_counter*>", "nbr_state_change_counter": "<*nbr_state_change_counter*>", "nbr_bad_lsreqs_counter": "<*nbr_bad_lsreqs_counter*>", "nbr_interval_expired_counter": "<*nbr_interval_expired_counter*>", "nbr_seq_number_mismatch": "<*nbr_seq_number_mismatch*>", "spf_full": "<*spf_full*>", "spf_summary": "<*spf_summary*>", "spf_external": "<*spf_external*>", "recv_buf": "<*recv_buf*>", "send_buf": "<*send_buf*>", "lsa_buf": "<*lsa_buf*>", "packet_unuse": "<*packet_unuse*>", "packet_max": "<*packet_max*>", "lsa_unuse": "<*lsa_unuse*>", "lsa_max": "<*lsa_max*>", "routerLsa_generated": "<*routerLsa_generated*>", "routerLsa_refreshed":"<*routerLsa_refreshed*>", "routerLsa_flushed": "<*routerLsa_flushed*>", "routerLsa_agedOut": "<*routerLsa_agedOut*>", "networkLsa_generated": "<*networkLsa_generated*>", "networkLsa_refreshed": "<*networkLsa_refreshed*>", "networkLsa_flushed": "<*networkLsa_flushed*>", "networkLsa_agedOut": "<*networkLsa_agedOut*>", "summaryLsa_generated": "<*summaryLsa_generated*>", "summaryLsa_refreshed": "<*summaryLsa_refreshed*>", "summaryLsa_flushed": "<*summaryLsa_flushed*>", "summaryLsa_agedOut": "<*summaryLsa_agedOut*>", "asbrSummaryLsa_generated": "<*asbrSummaryLsa_generated*>", "asbrSummaryLsa_refreshed": "<*asbrSummaryLsa_refreshed*>", "asbrSummaryLsa_flushed": "<*asbrSummaryLsa_flushed*>", "asbrSummaryLsa_agedOut": "<*asbrSummaryLsa_agedOut*>", "asExternalLsa_generated": "<*asExternalLsa_generated*>",

Response Body (JSON)	"asExternalLsa_refreshed": "<*asExternalLsa_refreshed*>", "asExternalLsa_flushed": "<*asExternalLsa_flushed*>", "asExternalLsa_agedOut": "<*asExternalLsa_agedOut*>", "asNssaLsa_generated": "<*asNssaLsa_generated*>", "asNssaLsa_refreshed": "<*asNssaLsa_refreshed*>", "asNssaLsa_flushed": "<*asNssaLsa_flushed*>", "asNssaLsa_agedOut": "<*asNssaLsa_agedOut*>", "type8Lsa_generated": "<*type8Lsa_generated*>", "type8Lsa_refreshed": "<*type8Lsa_refreshed*>", "type8Lsa_flushed": "<*type8Lsa_flushed*>", "type8Lsa_agedOut": "<*type8Lsa_agedOut*>", "linkOpaqueLsa_generated": "<*linkOpaqueLsa_generated*>", "linkOpaqueLsa_refreshed": "<*linkOpaqueLsa_refreshed*>", "linkOpaqueLsa_flushed": "<*linkOpaqueLsa_flushed*>", "linkOpaqueLsa_agedOut": "<*linkOpaqueLsa_agedOut*>", "areaOpaqueLsa_generated": "<*areaOpaqueLsa_generated*>", "areaOpaqueLsa_refreshed": "<*areaOpaqueLsa_refreshed*>", "areaOpaqueLsa_flushed": "<*areaOpaqueLsa_flushed*>", "areaOpaqueLsa_agedOut": "<*areaOpaqueLsa_agedOut*>", "asOpaqueLsa_generated": "<*asOpaqueLsa_generated*>", "asOpaqueLsa_refreshed": "<*asOpaqueLsa_refreshed*>", "asOpaqueLsa_flushed": "<*asOpaqueLsa_flushed*>", "asOpaqueLsa_agedOut": "<*asOpaqueLsa_agedOut*>" }, "vrf_name": "<*vrf_name*>" }]

where:

Element	Description
ospf_id	OSPF identifier. Default value: *0*.
clr_timer_str	Time since last OSPF process clear in the following format: HH:MM:SS.
router_id_changes	Router-ID changes counter; a positive integer.
dr_election_ counter	DR elections counter; a positive integer.
older_lsas_counter	Older received LSAs counter; a positive integer.
nbr_state_change_ counter	Neighbor state changes counter; a positive integer.
nbr_bad_lsreqs_ counter	Neighbor bad LS received requests counter; a positive integer.
nbr_interval_ expired_counter	Neighbor dead-interval expirations counter; a positive integer.
nbr_seq_number_ mismatch	Neighbor sequence number mismatches counter; a positive integer.
spf_full	Full SPF Computations counter; a positive integer.

Element	Description
spf_summary	Summary SPF Computations counter; a positive integer.
spf_external	External SPF Computations counter; a positive integer.
recv_buf	Received packet buffer; a positive integer.
send_buf	Sent packet buffer; a positive integer.
lsa_buf	LSA buffer; a positive integer.
packet_unuse	Unused packets number; a positive integer.
packet_max	Maximum packets number; a positive integer.
lsa_unuse	Unused LSAs number; a positive integer.
lsa_max	Maximum LSAs number; a positive integer.
router_lsa_type	Router LSA type name; a positive integer.
routerLsa_ generated	Number of generated router LSAs; a positive integer.
routerLsa_ refreshed	Number of refreshed router LSAs; a positive integer.
routerLsa_flushed	Number of flushed router LSAs; a positive integer.
routerLsa_agedOut	Number of aged out router LSAs; a positive integer.
networkLsa_ generated	Number of generated network LSAs; a positive integer.
networkLsa_ refreshed	Number of refreshed network LSAs; a positive integer.
networkLsa_flushed	Number of flushed network LSAs; a positive integer.
networkLsa_agedOut	Number of aged out network LSAs; a positive integer.
summaryLsa_ generated	Number of generated summary LSAs; a positive integer.
summaryLsa_ refreshed	Number of refreshed summary LSAs; a positive integer.
summaryLsa_flushed	Number of flushed summary LSAs; a positive integer.
summaryLsa_agedOut	Number of aged out summary LSAs; a positive integer.
asbrSummaryLsa_ generated	Number of generated ASBR summary LSAs; a positive integer.
asbrSummaryLsa_ refreshed	Number of refreshed ASBR summary LSAs; a positive integer.

Element	Description
asbrSummaryLsa_flushed	Number of flushed ASBR summary LSAs; a positive integer.
asbrSummaryLsa_agedOut	Number of aged out ASBR summary LSAs; a positive integer.
asExternalLsa_generated	Number of generated AS-External LSAs; a positive integer.
asExternalLsa_refreshed	Number of refreshed AS-External LSAs; a positive integer.
asExternalLsa_flushed	Number of flushed AS-External LSAs; a positive integer.
asExternalLsa_agedOut	Number of aged out AS-External LSAs; a positive integer.
asNssaLsa_generated	Number of generated AS-NSSA LSAs; a positive integer.
asNssaLsa_refreshed	Number of refreshed AS-NSSA LSAs; a positive integer.
asNssaLsa_flushed	Number of flushed AS-NSSA LSAs; a positive integer.
asNssaLsa_agedOut	Number of aged out AS-NSSA LSAs; a positive integer.
type8Lsa_generated	Number of generated type-8 LSAs; a positive integer.
type8Lsa_refreshed	Number of refreshed type-8 LSAs; a positive integer.
type8Lsa_flushed	Number of flushed type-8 LSAs; a positive integer.
type8Lsa_agedOut	Number of aged out type-8 LSAs; a positive integer.
linkOpaqueLsa_generated	Number of generated Link Opaque LSAs; a positive integer.
linkOpaqueLsa_refreshed	Number of refreshed Link Opaque LSAs; a positive integer.
linkOpaqueLsa_flushed	Number of flushed Link Opaque LSAs; a positive integer.
linkOpaqueLsa_agedOut	Number of aged out Link Opaque LSAs; a positive integer.
areaOpaque_lsa_type	Area Opaque LSA type name; a positive integer.
areaOpaqueLsa_generated	Number of generated Area Opaque LSAs; a positive integer.
areaOpaqueLsa_refreshed	Number of refreshed Area Opaque LSAs; a positive integer.

Element	Description
areaOpaqueLsa_flushed	Number of flushed Area Opaque LSAs; a positive integer.
areaOpaqueLsa_agedOut	Number of aged out Area Opaque LSAs; a positive integer.
asOpaque_lsa_type	AS Opaque LSA type name; a positive integer.
asOpaqueLsa_generated	Number of generated AS External Opaque LSAs; a positive integer.
asOpaqueLsa_refreshed	Number of refreshed AS External Opaque LSAs; a positive integer.
asOpaqueLsa_flushed	Number of flushed AS External Opaque LSAs; a positive integer.
asOpaqueLsa_agedOut	Number of aged out AS External Opaque LSAs; a positive integer.
vrf_name	Default VRF name. Default value: *default*.

Get OSPF Traffic Statistics

Gets the OSPF traffic statistics.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/traffic-stats/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name.Default value: `default`.

Response

Response Body

(JSON)

```
[
    {
        "traffic-stats":
        {
            "ospf_id": "<ospf_id>",
            "timer_str": "<timer_str>",
            "total_pkt_in": "<total_pkt_in>",
            "total_pkt_out":"<total_pkt_out>",
            "hello_in": "<hello_in>",
            "hello_out": "<hello_out>",
            "db_desc_in": "<db_desc_in>",
            "db_desc_out": "<db_desc_out>",
            "ls_req_in": "<ls_req_in>",
            "ls_req_out": "<ls_req_out>",
            "ls_upd_in": "<ls_upd_in>",
            "ls_upd_out": "<ls_upd_out>",
            "ls_ack_in": "<ls_ack_in>",
            "ls_ack_out": "<ls_ack_out>",
            "error_drops_in": "<error_drops_in>",
            "error_drops_out": "<error_drops_out>",
            "error_hellosin": "<error_hellosin>",
            "error_dbsin": "<error_dbsin>",
            "error_lsreqin": "<error_lsreqin>",
            "error_lsuin": "<error_lsuin>",
            "error_lsackin": "<error_lsackin>",
            "error_unknown_in": "<error_unknown_in>",
            "error_unknown_out": "<error_unknown_out>",
            "error_badcrc": "<error_badcrc>",
            "error_wrong_area": "<error_wrong_area>",
            "error_bad_version": "<error_bad_version>",
            "error_bad_auth": "<error_bad_auth>",
            "error_passive": "<error_passive>",
            "error_nonbr": "<error_nonbr>",
            "error_invalid_src": "<error_invalid_src>",
            "error_invalid_dst": "<error_invalid_dst>",
            "error_pktlength": "<error_pktlength>"
        },
        "vrf_name": "<vrf_name>"
    }
]
```

where:

Element	Description
`ospf_id`	OSPF identifier. Default value: *0*.
`timer_str`	Time since last OSPF process clear in the following format: HH:MM:SS.
`total_pkt_in`	Number of total packets in; a positive integer.
`total_pkt_out`	Number of total packets out; a positive integer.
`hello_in`	Number of hello packets in; a positive integer.
`hello_out`	Number of hello packets out; a positive integer.
`db_desc_in`	Number of DB descriptor packets in; a positive integer.

Element	Description
db_desc_out	Number of DB descriptor packets out; a positive integer.
ls_req_in	Number of LS Request packets in; a positive integer.
ls_req_out	Number of LS Request packets out; a positive integer.
ls_upd_in	Number of LS Update packets in; a positive integer.
ls_upd_out	Number of LS Update packets out; a positive integer.
ls_ack_in	Number of LS ACK packets in; a positive integer.
ls_ack_out	Number of LS ACK packets out; a positive integer.
error_drops_in	Number of errors related to drops in; a positive integer.
error_drops_out	Number of errors related to drops out; a positive integer.
error_hellosin	Number of errors related to hellos in; a positive integer.
error_dbsin	Number of errors related to DB Descriptors; a positive integer.
error_lsreqin	Number of errors related to LS Requests; a positive integer.
error_lsuin	Number of errors related to LS Updates; a positive integer.
error_lsackin	Number of errors related to LS ACKs; a positive integer.
error_unknown_in	Number of errors related to unknown in; a positive integer.
error_unknown_out	Number of errors related to unknown out; a positive integer.
error_badcrc	Number of errors related to Bad CRC; a positive integer.
error_wrong_area	Number of errors related to Wrong Area; a positive integer.
error_bad_version	Number of errors related to Bad Version; a positive integer.
error_bad_auth	Number of errors related to Bad Authentication; a positive integer.
error_passive	Number of errors related to Passive; a positive integer.
error_nonbr	Number of errors related to No Neighbor; a positive integer.
error_invalid_src	Number of errors related to Invalid Source; a positive integer.
error_invalid_dst	Number of errors related to Invalid Destination; a positive integer.

Element	Description
`error_pktlength`	Number of errors related to Packet Length; a positive integer.
`vrf_name`	Default VRF name. Default value: *default*.

Get OSPF Neighbors

Gets the OSPF neighbors list.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/neighbor/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body

(JSON)

```
[
   {
        "vrf_name": "<vrf_name>",
        "nbr_router_id": "<nbr_router_id>",
        "priority": "<priority>",
        "nbr_state": "<nbr_state>",
        "dead_timer": "<dead_timer>",
        "nbr_addr": "<nbr_addr>",
        "ifp_name": "<ifp_name>"
    }
]
```

where:

Element	Description
`vrf_name`	Default VRF name. Default value: *default*.
`nbr_router_id`	Neighbor router ID identifier; a valid IPv4 or IPv6 address.
`priority`	The neighbor priority; an integer from *0-255*.
`dead_timer`	The time left for dead interval expiry in the following format: HH:MM:SS.
`nbr_addr`	Neighbor IP address; a valid IPv4 or IPv6 address.
`ifp_name`	Ethernet interface name.

Get OSPF Routes

Gets the OSPF routes list.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/route/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body

(JSON)

```
[
    {
        "Network": "<Network>",
        "pathcode": "<pathcode>",
        "pathCount": "<pathCount>",
        "route_path_cost": "<route_path_cost>",
        "route_type2path_cost": "<route_type2path_cost>",
        "next_hop_info":
        [
           {
               "interface": "<interface>",
               "area_id": "<area_id>",
               "neighbor_addr": "<neighbor_addr>"
           }
        ]
    }
]
```

where:

Element	Description
`network`	Network name; a string in the following format: "AA:BB:CC:DD/MM".
`pathcode`	Path type; one of: • *connected* • *Discard* • *OSPF* • *OSPF inter area* • *OSPF NSSA external type 1* • *OSPF NSSA external type 2* • *OSPF external type 1* • *OSPF external type 2*
`pathCount`	Number of ecmp paths; a positive integer.
`route_path_cost`	Route-path cost; a positive integer.
`route_type2path_cost`	Route-type 2 path cost; a positive integer.
`next_hop_info`	Next-hop information; a list of dictionaries. Depending on the configuration, each dictionary may contain the following values: • *interface:* Neighbor IP address; a valid IPv4 or IPv6 address. • *area_id:* Neighbor area-ID; a valid IPv4 or IPv6 address. • *neighbor_addr:* Neighbor IP address; a valid IPv4 or IPv6 address.

Get OSPF Database

Gets the OSPF database.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/database/*<vrf_name>*
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "link_state_id": "`*<link_state_id>*`",`
	`  "adv_router": "`*<adv_router>*`",`
	`  "lsa_type": "`*<lsa_type>*`",`
	`  "lsa_age": "`*<lsa_age>*`",`
	`  "ls_seqnum_str": "`*<ls_seqnum_str>*`",`
	`  "checksum": "`*<checksum>*`",`
	`  "link count": "`*<link count>*`",`
	`  "area_id": "`*<area_id>*`"`
	`  }`
	`]`

where:

Element	Description
`link_state_id`	VRF name; a valid IPv4 or IPv6 address.
`adv_router`	Advertising router ID; a valid IPv4 or IPv6 address.
`lsa_type`	LSA type; one of: • *Router-LSA* • *Network-LSA* • *Summary-LSA* • *ASBR-summary-LSA* • *AS-external-LSA* • *AS-NSSA-LSA*
`lsa_age`	LSA age; a positive integer.

Element	Description
`ls_seqnum_str`	LS sequence number in hexadecimal format.
`checksum`	LSA checksum in hexadecimal format.
`link count`	Links number; a positive integer.
`area_id`	The area-ID of the LSDB; a valid IPv4 or IPv6 address.

Get OSPF Border Routers

Gets the OSPF border routers.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/border-router/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body

(JSON)

```
[
    {
            "abr_id": "<abr_id>",
            "abr_route_type":"<abr_route_type>",
            "abr_route_metric":"<abr_route_metric>",
            "asbr_id": "<asbr_id }",
            "asbr_route_type":"<asbr_route_type>",
            "asbr_route_metric":"<asbr_route_metric>,
            "type_border_router": "<type_border_router>",
            "abr_via": "<abr_via>",
            "asbr_via": "<asbr_via>",
            "abr_transit_area": "<abr_transit_area>",
            "asbr_transit_area": "<asbr_transit_area>",
            "abr_area_ifname": "<abr_area_ifname>",
            "asbr_area_ifname": "<asbr_area_ifname>"
     }
]
```

where:

Element	Description
`abr_id`	The ABR ID (string); shows the type, router ID or cost.
`abr_route_type`	Type of router related to ABR (string).
`abr_route_metric`	Metric of router related to ABR (string).
`asbr_id`	The ASBR ID (string); shows the type, router ID or cost.
`asbr_route_type`	Type of router related to ASBR (string).
`asbr_route_metric`	Metric of router related to ASBR (string).

Element	Description
type_border_router	The border router type (string); one of ABR or ASBR.
abr_via	The next-hop IP for ABR (string); a valid IP address.
asbr_via	The next-hop IP for ABSBR (string); a valid IP address.
abr_transit_area	The transit area-ID for ABR (string); a valid IP address.
asbr_transit_area	The transit area-ID for ASBR (string); a valid IP address.
abr_area_ifname	The OSPF interface for ABR (string). For example: *Ethernet1/X* or *VLAN interface*.
asbr_area_ifname	The OSPF interface for ABSR (string). For example: *Ethernet1/X* or *VLAN interface*.

Get OSPF Summary Address

Gets the OSPF summary address.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/summary-address/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`{`
	`"router_id": "`<*router_id*>`",`
	`"ospf_id": "`<*ospf_id*>`",`
	`"vrf_name": "`<*vrf_name*>`",`
	`"prefix": "`<*prefix*>`",`
	`"metric":"`<*metric*>`",`
	`"tag": "`<*tag*>`",`
	`"summary_address_state":"`<*summary_address_state*>`"`
	`}`
	`]`

where::

Element	Description
`router_id`	Router-ID in IP address format (string); a valid IP address.
`ospf_id`	OSPF identifier (integer). Default value: *0*.
`vrf_name`	Default VRF name (string). Default value: *default*.
`prefix`	The IP prefix (string), in the following format: *XX.XX.XX.XX/XX*.
`metric`	The metric value; an integer from *0-16777214*.
`tag`	External/NSSA LSAs tag; a positive integer from *0-4294967295*.
`summary_address_state`	The summary address status (string); one of *Active, Pending*.

Get OSPF Interface

Displays the OSPF interface information.

Request

Method Type	GET
Request URI	nos/api/cfg/ospf/interface/<*if_name*>/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The interface name.
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body
 	
(JSON)
	{
	"if_name": "*<ip_name>*",
	"vrf_name": "*<vrf_name>*",
	"ospf_id": "*<ospf_id>*",
	"ospf_status": "*<ospf_status>*",
	"if_addr": "*<if_addr>*",
	"if_area_id": "*<if_area_id>*",
	"if_mtu": "*<if_mtu>*",
	"router_id": "*<router_id>*",
	"if_network_type": "*<if_network_type>*",
	"if_output_cost": "*<if_output_cost>*",
	"if_transmit_delay": "*<if_transmit_delay>*",
	"priority": "*<priority>*",
	"if_state": "*<if_state>*",
	"designated_router":"*<designated_router>*",
	"designated_router_addr":"*<designated_router_addr>*",
	"backup_designated_router":"*<backup_designated_router>*",
	"backup_designated_router_addr": "*<backup_designated_router_addr>*",
	"hello_interval": "*<hello_interval>*",
	"dead_interval": "*<dead_interval>*",
	"retransmit_interval": "*<retransmit_interval>*",
	"if_hello_timer": "*<if_hello_timer>*"
	"neighbor_count": "*<neighbor_count>*",
	"adj_neighbor_count": "*<adj_neighbor_count>*",
	"hello_in": "*<hello_in>*",
	"hello_out": "*<hello_out>*",
	"ls_req_in": "*<ls_req_in>*",
	"ls_req_out": "*<ls_req_out>*",
	"ls_upd_in": "*<ls_upd_in>*"
	"ls_upd_out": "*<ls_upd_out>*",
	"ls_ack_in": "*<ls_ack_in>*",
	"ls_ack_out": "*<ls_ack_out>*",
	"db_desc_in": "*<db_desc_in>*",
	"db_desc_out": "*<db_desc_out>*",
	"discarded": "*<discarded>*",
	"if_mtu_ignore": "*<if_mtu_ignore>*",
	"passive_interface": "*<passive_interface>*",
	"if_bfd": "*<if_bfd>*",
	"db_filter_all_out": "*<db_filter_all_out>*",
	"auth_type": "*<auth_type>*",
	"key_id": "*<key_id>*"
	}

where:

Element	Description
`if_name`	The interface name (string). For example: *Ethernet1/X* or *VLAN interface*.
`vrf_name`	Default VRF name (string). Default value: `default`.
`ospf_id`	OSPF process identifier (integer). Default value: `0`.
`ospf_status`	The status of the OSPF protocol (string); one of `Up`, `Down`.
`if_addr`	The IP address or mask (string); a valid IP address or mask.

Element	Description
`if_area_id`	The area-ID (string); a valid IP address.
`if_mtu`	The maximum transmission unit; a positive integer from *576-65535*.
`router_id`	The router-ID in IP address format (string); a valid IP address.
`if_network_type`	The network type (string); one of *Broadcast, Point-to-Point*.
`if_output_cost`	Interface output cost; a positive integer from *1-65535*.
`if_transmit_delay`	The interface transmit delay, in seconds; an integer from *1-3600*.
`priority`	The router priority; an integer from *0-255*.
`if_state`	The operation state of the interface (string); one of *DR, Backup, DRother*.
`designated_router`	Designated Router-ID (string); a valid IP address.
`designated_router_addr`	The IP address for the designated router (string).
`backup_designated_router`	The backup Router-ID for the designated router (string); a valid IP address.
`backup_designated_router_addr`	The backup Router-ID for the designated router. (string).
`hello_interval`	The hello interval, in seconds; an integer from *1-65535*.
`dead_interval`	The dead interval, in seconds; an integer from *1-65535*.
`retransmit_interval`	The retransmit interval, in seconds; an integer from *1-65535*.
`if_hello_timer`	The hello interval timer expiration time (string).
`neighbor_count`	The neighbor count (integer); a positive integer.
`adj_neighbor_count`	The adjacent neighbors count (integer); a positive integer.
`hello_in`	Number of total hello packets in; a positive integer.
`hello_out`	Number of total hello packets out; a positive integer.
`ls_req_in`	Number of total LS Request packets in; a positive integer.
`ls_req_out`	Number of total LS Request packets out; a positive integer.

Element	Description
ls_upd_in	Number of total LS Update packets in; a positive integer.
ls_upd_out	Number of total LS Update packets out; a positive integer.
ls_ack_in	Number of total LS ACK packets in; a positive integer.
ls_ack_out	Number of total LS ACK packets out; a positive integer.
db_desc_in	Number of total DB Descriptors packets in; a positive integer.
db_desc_out	Number of total DB Descriptors packets out; a positive integer.
discarded	Number of total discarded packets; a positive integer.
auth_type	The type of authentication; one of *Message-Digest, Simple, Null*.
key_id	The Key-ID, if the authentication type is MD5/SHA256; an integer from *1-255*.
if_mtu_ignore	The maximum transmission unit status; one of *Enable, Disable*.
passive_interface	The passive interface status; one of *Enable, Disable*.
if_bfd	The BDF status; one of *Enable, Disable*.
db_filter_all_out	Database filter all out; one of *Enable, Disable*.

Set OSPF Interface

Sets the OSPF interface.

Request

Method Type	PUT
Request URI	nos/api/cfg/ospf/interface/<*ifname*>/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"ospf_status": "<*ospf_status*>",
	"if_area_id": "<*if_area_id*>",
	"if_mtu": <*if_mtu*>,
	"if_network_type": "<*if_network_type*>",
	"if_output_cost": "<*if_output_cost*>",
	"if_transmit_delay": "<*if_transmit_delay*>",
	"priority": "<*priority*>",
	"hello_interval": "<*hello_interval*>",
	"dead_interval": "<*dead_interval*>",
	"retransmit_interval": "<*retransmit_interval*>",
	"if_mtu_ignore": "<*if_mtu_ignore*>",
	"passive_interface": "<*passive_interface*>",
	"if_bfd": "<*if_bfd*>",
	"db_filter_all_out": "<*db_filter_all_out*>",
	"auth_type": "<*auth_type*>",
	"auth_key": "<*auth_key*>",
	"key_id": "<*key_id*>",
	"md5 ": "<*md5_password*>",or "sha256":"<*sha256_password*>",
	"set_to_default": [set_to_default],
	"key_remove_with_key_id": "<*key_remove_with_key_id*>"
	}

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`if_name`	The interface name (string). For example: *Ethernet1/X* or *VLAN interface*.
`ospf_status`	The status of the OSPF protocol (string); one of *Up, Down*.
`if_area_id`	The area-ID (string); a valid IP address.
`if_mtu`	The MTU size, from *576-65535*.
`if_network_type`	The network type (string); one of *Broadcast, Point-to-Point*.
`if_output_cost`	Interface output cost; a positive integer from *1-65535*.
`if_transmit_delay`	The interface transmit delay, in seconds; an integer from *1-3600*.
`priority`	The router priority; an integer from *0-255*.

Element	Description
`hello_interval`	The hello interval, in seconds; an integer from *1-65535*.
`dead_interval`	The dead interval, in seconds; an integer from *1-65535*.
`retransmit_interval`	The retransmit interval, in seconds; an integer from *1-65535*.
`auth_type`	The type of authentication; one of *Message-Digest, Simple, Null*.
`auth_key`	The key of authentication.
`key_id`	The Key-ID, if the authentication type is MD5/SHA256; an integer from *1-255*.
`if_mtu_ignore`	The maximum transmission unit status; one of *Enable, Disable*.
`passive_interface`	The passive interface status; one of *Enable, Disable.*
`if_bfd`	The BDF status; one of *Enable, Disable*.
`db_filter_all_out`	Database filter all out; one of *Enable, Disable*.
`md5/sha256`	The password.
`set_to_default`	The list of keys for which the configuration is set to the default settings: • *auth_key* • *auth_type* • *hello_interval* • *dead_interval* • *if_transmit_delay* • *retransmit_interval* • *if_output_cost, priority* • *if_mtu*
`key_remove_with_key_id`	The MD5 or SHA Key ID to be removed, from *1-256*.

Response

Response Body
 	
(JSON)
	{
	"if_name": "*<ip_name>*",
	"vrf_name": "*<vrf_name>*",
	"ospf_id": "*<ospf_id>*",
	"ospf_status": "*<ospf_status>*",
	"if_addr": "*<if_addr>*",
	"if_area_id": "*<if_area_id>*",
	"if_mtu": "*<if_mtu>*",
	"router_id": "*<router_id>*",
	"if_network_type": "*<if_network_type>*",
	"if_output_cost": "*<if_output_cost>*",
	"if_transmit_delay": "*<if_transmit_delay>*",
	"priority": "*<priority>*",
	"if_state": "*<if_state>*",
	"designated_router":"*<designated_router>*",
	"designated_router_addr":"*<designated_router_addr>*",
	"backup_designated_router":"*<backup_designated_router>*",
	"backup_designated_router_addr": "*< backup_designated_router_addr>*",
	"hello_interval": "*<hello_interval>*",
	"dead_interval": "*<dead_interval>*",
	"retransmit_interval": "*<retransmit_interval>*",
	"if_hello_timer": "*<if_hello_timer>*"
	"neighbor_count": "*<neighbor_count>*",
	"adj_neighbor_count": "*<adj_neighbor_count>*",
	"hello_in": "*<hello_in>*",
	"hello_out": "*<hello_out>*",
	"ls_req_in": "*<ls_req_in>*",
	"ls_req_out": "*<ls_req_out>*",
	"ls_upd_in": "*<ls_upd_in>*"
	"ls_upd_out": "*<ls_upd_out>*",
	"ls_ack_in": "*<ls_ack_in>*",
	"ls_ack_out": "*<ls_ack_out>*",
	"db_desc_in": "*<db_desc_in>*",
	"db_desc_out": "*<db_desc_out>*",
	"discarded": "*<discarded>*",
	"if_mtu_ignore": "*<if_mtu_ignore*>",
	"passive_interface": "*<passive_interface>*",
	"if_bfd": "*<if_bfd>*",
	"db_filter_all_out": "*<db_filter_all_out>*",
	"auth_type": "*<auth_type>*",
	"key_id": "*<key_id>*"
	}

where:

Element	Description
`if_name`	The interface name (string). For example: *Ethernet1/X* or *VLAN interface*.
`vrf_name`	Default VRF name (string). Default value: `default`.
`ospf_id`	OSPF process identifier (integer). Default value: `0`.
`ospf_status`	The status of the OSPF protocol (string); one of `Up`, `Down`.

Element	Description
`if_addr`	The IP address or mask (string); a valid IP address or mask.
`if_area_id`	The area-ID (string); a valid IP address.
`if_mtu`	The maximum transmission unit; a positive integer from *576-65535*.
`router_id`	The router-ID in IP address format (string); a valid IP address.
`if_network_type`	The network type (string); one of *Broadcast, Point-to-Point*.
`if_output_cost`	Interface output cost; a positive integer from *1-65535*.
`if_transmit_delay`	The interface transmit delay, in seconds; an integer from *1-3600.*
`priority`	The router priority; an integer from *0-255*.
`if_state`	The operation state of the interface (string); one of *DR, Backup, DRother*.
`designated_router`	Designated Router-ID (string); a valid IP address.
`designated_router_addr`	The IP address for the designated router (string).
`backup_designated_router`	The backup Router-ID for the designated router (string); a valid IP address.
`backup_designated_router_addr`	The backup Router-ID for the designated router. (string).
`hello_interval`	The hello interval, in seconds; an integer from *1-65535*.
`dead_interval`	The dead interval, in seconds; an integer from *1-65535*.
`retransmit_interval`	The retransmit interval, in seconds; an integer from *1-65535*.
`if_hello_timer`	The hello interval timer expiration time (string).
`neighbor_count`	The neighbor count (integer); a positive integer.
`adj_neighbor_count`	The adjacent neighbors count (integer); a positive integer.
`hello_in`	Number of total hello packets in; a positive integer.
`hello_out`	Number of total hello packets out; a positive integer.
`ls_req_in`	Number of total LS Request packets in; a positive integer.

Element	Description
`ls_req_out`	Number of total LS Request packets out; a positive integer.
`ls_upd_in`	Number of total LS Update packets in; a positive integer.
`ls_upd_out`	Number of total LS Update packets out; a positive integer.
`ls_ack_in`	Number of total LS ACK packets in; a positive integer.
`ls_ack_out`	Number of total LS ACK packets out; a positive integer.
`db_desc_in`	Number of total DB Descriptors packets in; a positive integer.
`db_desc_out`	Number of total DB Descriptors packets out; a positive integer.
`discarded`	Number of total discarded packets; a positive integer.
`auth_type`	The type of authentication; one of *Message-Digest, Simple, Null*.
`key_id`	The Key-ID, if the authentication type is MD5/SHA256; an integer from *1-255*.
`if_mtu_ignore`	The maximum transmission unit status; one of `Enable`, `Disable`.
`passive_interface`	The passive interface status; one of *Enable, Disable*.
`if_bfd`	The BDF status; one of *Enable, Disable.*
`db_filter_all_out`	Database filter all out; one of *Enable, Disable*.

Get OSPF Virtual Links

Gets the OSPF virtual-links.

Request

Method Type	GET
Request URI	/nos/api/cfg/ospf/virtual-link/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body

(JSON)

```
[
    {
        "vrf_name": "<vrf_name>",
        "vlink_name": "<vlink_name>",
        "nbr_router_id": "<nbr_router_id>",
        "ifp_name": "<ifp_name>",
        "local_address": "<local_address>",
        "remote_address": "<remote_address>",
        "transit_area": "<transit_area>",
        "transmit_delay": "<transmit_delay>",
        "vlink_state": "<vlink_state>",
        "hello_interval": "<hello_interval>",
        "dead_interval": "<dead_interval>",
        "wait_interval": "<wait_interval>",
        "retransmit_interval": "<retransmit_interval>",
        "hello_due": "<hello_due>",
        "adjacency_state": "<adjacency_state>",
        "authentication_type": "<authentication_type>",
        "key_id": "<key_id>"
    }
]
```

where:

Element	Description
`vrf_name`	Default VRF name. Default value: *default*.
`vlink_name`	The virtual-link name (string).
`nbr_router_id`	The neighbor router ID (string); a valid IP address.
`ifp_name`	The interface name (string). For example: *Ethernet1/X* or *VLAN interface*.

Element	Description
`local_address`	The local interface IP address (string); a valid IP address.
`remote_address`	The remote interface IP address (string); a valid IP address.
`transit_area`	The transit area-ID (string); a valid IP address.
`transmit_delay`	The transmission delay interval, in seconds, an integer from *1-3600*.
`vlink_state`	The Virtual Link status (string); one of *Up, Down*.
`hello_interval`	The hello interval, in seconds; an integer from *1-65535*.
`dead_interval`	The dead interval, in seconds; an integer from *1-65535*.
`wait_interval`	The wait interval, in seconds; an integer from *1-65535*.
`retransmit_interval`	The retransmit interval, in seconds; an integer from *1-65535*.
`hello_due`	The due time to send the next hello (string).
`adjacency_state`	The adjacency state across the virtual-link (string).
`authentication_type`	The type of authentication (string); one of *Message-Digest, Simple, Null*.
`key_id`	The Key-ID, if the authentication type is MD5/SHA256; an integer from *1-255*.

Set OSPF Virtual Links

Sets the OSPF virtual-links.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/virtual-link/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"vlink_name": "<*vlink_name*>",
	"area_id": "<*area_id*>",
	"nbr_router_id": "<*nbr_router_id*>",
	"virtual_link_disable": "< *virtual_link_disable*>",
	"hello_interval": <*hello_interval*>,
	"dead_interval": <*dead_interval*>,
	"retransmit_interval": < *retransmit_interval*>,
	"transmit_delay": <*transmit_delay*>
	"auth_type": "<*auth_type*>",
	"auth_key": "<*auth_key*>",
	"key_id": "<*key_id*>",
	"md5": "<*md5_password*>",or "sha256":"<*sha256_password*>",
	"bfd":"<*bfd*>",
	"set_to_default": [set_to_default],
	"key_remove_with_key_id": "<*key_remove_with_key_id*>"
	}

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`vlink_name`	The virtual-link name (string).
`area_id`	The area-ID (string); a valid IP address.
`nbr_router_id`	The neighbor router ID (string); a valid IP address.
`transmit_delay`	The transmission delay interval, in seconds, an integer from *1-3600*.
`virtual_link_disable`	The Virtual Link status (string); one of *Yes, No*.
`hello_interval`	The hello interval, in seconds; an integer from *1-65535*.
`dead_interval`	The dead interval, in seconds; an integer from *1-65535*.
`retransmit_interval`	The retransmit interval, in seconds; an integer from *1-65535*.
`bfd`	The BFD status; one of *Enable, Disable*.

Element	Description
auth_type	The type of authentication (string); one of *Message-Digest, Simple, Null*.
auth_key	The authentication key (string).
key_id	The Key-ID, if the authentication type is MD5/SHA256; an integer from *1-255*.
md5/sha256	The password for Message-digest Authentication.
set_to_default	The list of keys to set to default settings. One of: • *auth_key* • *auth_type* • *hello_interval* • *dead_interval* • *transmit_delay* • *retransmit_interval*
key_remove_with_key_id	The md5 key ID to remove.

Response

Response Body

(JSON)

```
[
    {
        "vrf_name": "<vrf_name>",
        "vlink_name": "<vlink_name>",
        "nbr_router_id": "<nbr_router_id>",
        "ifp_name": "<ifp_name>",
        "local_address": "<local_address>",
        "remote_address": "<remote_address>",
        "transit_area": "<transit_area>",
        "transmit_delay": "<transmit_delay>",
        "vlink_state": "<vlink_state>",
        "hello_interval": "<hello_interval>",
        "dead_interval": "<dead_interval>",
        "wait_interval": "<wait_interval>",
        "retransmit_interval": "<retransmit_interval>",
        "hello_due": "<hello_due>",
        "adjacency_state": "<adjacency_state>",
        "authentication_type": "<authentication_type>",
        "key_id": "<key_id>"
    }
]
```

where:

Element	Description
vrf_name	Default VRF name. Default value: *default*.
vlink_name	The virtual-link name (string).
nbr_router_id	The neighbor router ID (string); a valid IP address.

Element	Description
`ifp_name`	The interface name (string). For example: *Ethernet1/X* or *VLAN interface*.
`local_address`	The local interface IP address (string); a valid IP address.
`remote_address`	The remote interface IP address (string); a valid IP address.
`transit_area`	The transit area-ID (string); a valid IP address.
`transmit_delay`	The transmission delay interval, in seconds, an integer from *1-3600*.
`vlink_state`	The Virtual Link status (string); one of *Up, Down*.
`hello_interval`	The hello interval, in seconds; an integer from *1-65535*.
`dead_interval`	The dead interval, in seconds; an integer from *1-65535*.
`wait_interval`	The wait interval, in seconds; an integer from *1-65535*.
`retransmit_interval`	The retransmit interval, in seconds; an integer from *1-65535*.
`hello_due`	The due time to send the next hello (string).
`adjacency_state`	The adjacency state across the virtual-link (string).
`authentication_type`	The type of authentication (string); one of *Message-Digest, Simple, Null*.
`key_id`	The Key-ID, if the authentication type is MD5/SHA256; an integer from *1-255*.

Get OSPF Process

Gets the OSPF process information.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/process/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body

(JSON)

```
[
    {
        "ospfId": "<ospfId>",
        "routerId": "<routerId>",
        "upTime": "<upTime>",
        "vrfName": "<vrfName>",
        "vrfFlags":"<vrfFlags>",
        "abrType":"<abrType>",
        "spfStartDelaySec":"<spfStartDelaySec>",
        "spfStartDelayUsec":"<spfStartDelayUsec>",
        "spfMinDelaySec":"<spfMinDelaySec>",
        "spfMinDelayUsec":"<spfMinDelayUsec>",
        "spfMaxDelaySec":"<spfMaxDelaySec>",
        "lsdbCount":"<lsdbCount>",
        "lsdbChecksum":"<lsdbChecksum>",
        "originateNewLsas":"<originateNewLsas>",
        "rxNewLsas":"<rxNewLsas>",
        "distance_all":"<distance_all>",
        "distance_intra":"<distance_intra>",
        "distance_inter":"<distance_inter>",
        "distance_external":"<distance_external>",
        "area_info":
        [
           {
               "auth_type": "<auth_type>",
               "mode": "<mode>",
               "area_id": "<area_id>",
               "area_flags":"<area_flags>",
               "area_type":"<area_type>",
               "active_if_count":"<active_if_count>",
               "full_virt_nbr_count":"<full_virt_nbr_count>",
               "full_nbr_count":"<full_nbr_count>",
               "spf_calc_count":"<spf_calc_count>",
               "area_lsdb_count":"<area_lsdb_count>",
               "area_lsdb_checksum":"<area_lsdb_checksum>",
           }
        ]
    }
]
```

where:

Element	Description
`ospfId`	The OSPF process identifier (integer). Default value: *0*.
`routerId`	Router-ID in IP address format (string); a valid IP address.
`uptime`	The OSPF process uptime; a string in the following format: HH:MM:SS.
`vfrName`	Default VRF name (string). Default value: *default* **.**
`vrfFlags`	The VRF flag; a positive value.
`abrType`	The ABR type (integer); displays the valid ABR types.

Element	Description
`spfStartDelaySec`	The SPF schedule start delay, in seconds; an integer from *0-600*.
`spfStartDelayUsec`	The SPF schedule start delay, in microseconds; an integer from *0-1000*.
`spfMinDelaySec`	The minimum SPF schedule delay time; an integer from *1-600*.
`spfMinDelayUsec`	The minimum SPF schedule delay time, in microseconds; an integer from *1-1000*.
`lsdbCount`	The number of external LSAs; zero or a positive integer.
`lsdbChecksum`	LAS checksum value (integer).
`lsdbOverflow`	The number of LSAs exceeding the limit; zero or a positive integer.
`originateNewLsas`	The number of new originated LSAs; zero or a positive integer.
`rxNewLsas`	The number of new LSAs received; zero or a positive integer.
`distance_all`	The distance to all destinations; zero or a positive integer.
`distance_intra`	The distance to intra-area destinations; zero or a positive integer.
`distance_inter`	The distance to inter-area destinations; zero or a positive integer.
`distance_external`	The distance to external destinations; zero or a positive integer.
`auth_type`	The type of authentication (integer); one of *Null, zero or cryptographic*.
`mode`	The IS area shortcut (an integer); one of *Shortcut, none*.
`area_id`	The area-ID; an integer from *0-4294967295.*
`area_flags`	The area flag; a positive value.
`area_type`	The area type (integer); one of *Default, stub or nssa*.
`active_if_count`	The number of active interfaces in an area; zero or a positive integer.
`area_if_count`	The number of interfaces in an area; zero or a positive integer.
`full_virt_nbr_count`	Virtual neighbors count; zero or a positive integer.

Element	Description
`full_nbr_count`	Total number of neighbors; zero or a positive integer.
`spf_calc_count`	The number of SPF calculations; zero or a positive integer.
`area_lsdb_count`	The number of LSAs in the area; zero or a positive integer.
`area_lsbd_checksum`	The valid checksum of the link state database; a positive integer.

Get OSPF Multi-Area Neighbor

Gets the OSPF multi-area neighbor information.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/multiarea-neighbors/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body

(JSON)

```
[
    {
        "ifname": "<ifname>",
        "nbr_add": "<nbr_addr>",
        "type": "<type>",
        "ifIpAddress": "<ifIpAddress>",
        "ifAreaId": "<ifAreaId>",
        "ifMTU":"<ifMTU>",
        "proc_id":"<proc_id>",
        "ifRouterId":"<ifRouterId>",
        "ifTransmitDelay":"<ifTransmitDelay>",
        "ifNetworkType":"<ifNetworkType>",
        "if_output_cost":"<if_output_cost>",
        "d_router":"<d_router>",
        "D_router_address":"<D_router_address>",
        "Bd_router":"<Bd_router>",
        "Bd_router_address":"<Bd_router_address>",
        "hello_interval":"<hello_interval>",
        "dead_interval":"<dead_interval>",
        "retransmit_interval":"<retransmit_interval>",
        "neighbor_count":"<neighbor_count>"
    }
]
```

where:

Element	Description
`ifName`	The interface name (string). For example: *Ethernet1/X* or *VLAN interface*.
`nbr_addr`	Neighbor IP address (string); a valid IP address.
`type`	The area type (integer); one of *Default, stub* or *nssa*.

Element	Description
`ifIpAddress`	The interface IP address (string); a valid IP address.
`ifAreaId`	The interface area-ID; an integer from *0-4294967295*.
`ifMTU`	The maximum transmission unit (integer).
`proc_id`	The OSPF process identifier (integer). Default value: *0*.
`ifRouterId`	The router-ID in IP address format (string); a valid IP address.
`ifNetworkType`	The interface network type (integer). Default value: *Point-to-Point*.
`if_output_cost`	Interface output cost; zero or a positive integer.
`if_transmit_delay`	The interface transmit delay, in seconds (integer).
`transmit_if_state`	The interface state type (integer).
`d_router`	Designated Router-ID (string); a valid IP address.
`D_router_address`	The IP address for the designated router (string).
`Bd_router`	The backup Router-ID for the designated router (string); a valid IP address.
`Bd_router_address`	The backup address for the designated router. (string).
`Hello_interval`	The hello interval, in seconds (integer).
`Dead_interval`	The dead interval, in seconds (integer).
`retransmit_interval`	The retransmit interval, in seconds (integer).
`neighbor_count`	The number of multi-area adjacent neighbors (integer).

Get OSPF RIB Counters

Gets the OSPF RIB counters information.

Request

Method Type	GET
Request URI	/nos/api/info/ospf/ribcounter_info/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body

(JSON)

```
[
    {
        "ospf2rib_route_add": "<ospf2rib_route_add>",
        "ospf2rib_route_add_error": "<ospf2rib_route_add_error>",
        "ospf2rib_route_delete": "<ospf2rib_route_delete>",
        "ospf2rib_route_delete_error":"<ospf2rib_route_delete_error>",
        "ospf_route_adds": "<ospf_route_adds>",
        "ospf_route_dels": "<ospf_route_dels >",
        "ospf_route_adds_ignored":"<ospf_route_adds_ignored >",
        "ospf_route_dels_ignored":"<ospf_route_dels_ignored >,
        "rib2ospf_route_add":"<rib2ospf_route_add >",
        "rib2ospf_route_add_error":"<rib2ospf_route_add_error>",
        "rib2ospf_route_del":"<rib2ospf_route_del>",
        "rib2ospf_route_del_error":"<rib2ospf_route_del_error>",
    }
]
```

where:

Element	Description
`ospf2rib_route_add`	The OSPF route addition calls made to RIB (integer).
`ospf2rib_route_add_error`	The OSPF to RIB route addition call errors (integer).
`ospf2rib_route_delete`	The OSPF to RIB route deletion calls (integer).
`ospf2rib_route_delete_error`	The OSPF to RIB route deletion call errors (integer).
`ospf2rib_route_add`	The OSPF route addition calls (integer).
`ospf2rib_route_dels`	The OSPF route deleted calls (integer).

Element	Description
ospf_route_adds	The OSPF route addition calls (integer).
ospf_route_dels	The OSPF route deleted calls (integer).
ospf_route_adds_ignored	The OSPF ignored route addition calls (integer).
ospf_route_dels_ignored	The OSPF ignored route deleted calls (integer).
rib2ospf_route_add	The number of route additional calls from RIB to OSPF (integer).
rib2ospf_route_add_error	The number of route additional call errors from RIB to OSPF (integer).
rib2ospf_route_del	The number of route deleted calls from RIB to OSPF (integer).
rib2ospf_route_del_error	The number of route deleted call errors from RIB to OSPF (integer).

Set OSPF Process

Updates the OSPF process information.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/process/

Request Body
 	
(JSON)
	{
	"routerId": "*<routerId>*",
	"defaultMetric":"*<defaultMetric>*",
	"distance_all":"*<distance_all>*",
	"bfd":"*<bfd>*",
	"shutdown":"*<shutdown>*"
	}

where:

Variable	Description
routerId	The OPSF router-ID in IP address format (string); a valid IP address.
defaultMetric	The default metric cost; an integer from *1-16777214*.
distance_all	The administrative distance, an integer from 1-255.
bfd	Whether to enable BFD configuration (string); one of *Enable, Disable*.
shutdown	Whether to enable the shutdown OSPF process (string); one of *Enable, Disable*.

Response

Response Body

(JSON)

```
{
    "ospfId": "<ospfId>",
    "routerId": "<routerId>",
    "upTime": "<upTime>",
    "vrfName": "<vrfName>",
    "adminFlags":"<adminFlags>",
    "configFlags":"<configFlags>",
    "abrType":"<abrType>",
    "spfStartDelaySec":"<spfStartDelaySec>",
    "spfStartDelayUsec":"<spfStartDelayUsec>",
    "spfMinDelaySec":"<spfMinDelaySec>",
    "spfMinDelayUsec":"<spfMinDelayUsec>",
    "spfMaxDelaySec":"<spfMaxDelaySec>",
    "lsdbCount":"<lsdbCount>",
    "lsdbChecksum":"<lsdbChecksum>",
    "originateNewLsas":"<originateNewLsas>",
    "rxNewLsas":"<rxNewLsas>",
    "distance_all":"<distance_all>",
    "distance_intra":"<distance_intra>",
    "distance_inter":"<distance_inter>",
    "distance_external":"<distance_external>",
    "area_info":
    [
       {
           "auth_type": "<auth_type>",
           "mode": "<mode>",
           "area_id": "<area_id>",
           "area_flags":"<area_flags>",
           "area_type":"<area_type>",
           "active_if_count":"<active_if_count>",
           "full_virt_nbr_count":"<full_virt_nbr_count>",
           "full_nbr_count":"<full_nbr_count>",
           "spf_calc_count":"<spf_calc_count>",
           "area_lsdb_count":"<area_lsdb_count>",
           "area_lsdb_checksum":"<area_lsdb_checksum>",
       }
    ]
}
```

where:

Element	Description
`ospfId`	OSPF process ID. Default value: *0*.
`routerId`	The router ID; a valid IP address.
`upTime`	OSPF process uptime, in the following format: hh:mm:ss.
`vrfName`	The VRF name; a valid IP address.
`adminFlag`	Administrative Flags; a positive number.
`configFlag`	Configuration flags; a positive number.
`vrf flags`	VRF flags; a positive number.
`abrType`	Area Border Router type; a type of ABR.

Element	Description
spfStartDelaySec	SPF schedule start delay; the delay value, in seconds.
spfStartDelayUsec	SPF schedule start delay; the delay value, in microseconds.
spfMinDelayMinSec	The minimum SPF schedule delay time; the delay value, in seconds.
spfMinDelayMinUsec	The minimum SPF schedule delay time in microseconds; the delay value, in microseconds.
lsdbCount	Number of as-external-LSAs; zero, or a positive number.
lsdbChecksum	The Checksum value; a valid checksum value.
lsdbOverflow	Number of LSAs exceeding limit value; zero, or a positive number.
originateNewLsas	Number of new LSAs originated; zero, or a positive number.
rxNewLsas	Number of new LSAs received; zero or a positive number.
distance_all	Distances of all destinations; zero or a positive number.
distance_intra	Distance of intra-area destinations; zero or a positive number.
distance_inter	Distance of inter-area destinations; zero or a positive number.
distance_external	Distance of external destinations; zero or a positive number.
auth_type	Authentication type; one of *null, zero, cryptographic*.
mode is	Area configured as shortcut; one of *shortcut mode* or *none*.
area_id	The Area ID; zero or a positive number.
area_type	Type of area; one of *default, stub, nssa*.
active_if_count	Active interfaces in an area; zero or a positive number.
area_if_count	Number of interfaces in an area; zero or a positive number.
full_virt_nbr_count	Count of virtual neighbors; zero or a positive number.
full_nbr_count	Number of neighbors; zero or a positive number.
spf_calc_count	Number of SPF calculations; zero or a positive number.

Element	Description
`area_lsdb_count`	Number of LSAs in link state database; zero or a positive number.
`area_lsdb_checksum`	Checksum of link state database; a positive number.

Get OSPF Redistribute

Gets the OSPF redistribution.

Request

Method Type	GET
Request URI	/nos/api/cfg/ospf/redistribute/
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"vrfName": "*<vrfName>*",
	"redist_direct":"*<redist_direct>*",
	"direct_metric":"*<direct_metric>*",
	"direct_metric_type":"*<direct_metric_type>*",
	"direct_tag":"*<direct_tag>*",
	"direct_rmap_name":"*<direct_rmap_name>*",
	"redist_bgp":"*<redist_bgp>*",
	"bgp_metric":"*<bgp_metric>*",
	"bgp_metric_type": "*<bgp_metric>*",
	"bgp_tag":" "*<bgp_tag>*",
	"bgp_rmap_name":"*<bgp_rmap_name>*",
	"redist_static":"<redist_static>",
	"static_metric":"*<static_metric>*",
	"static_metric_type":"*<static_metric_type>*",
	"static_tag":"*<static_tag>*",
	"static_rmap_name":"*<static_rmap_name>*"
	}

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: *default*.
redist_direct	Redistribute the direct configuration (string); one of *Enable, Disable*.
direct_metric	Redistribute the direct cost; an integer from *0-16777214*.
direct_metric_type	The external metric type (integer); one of *1,2*.
direct_tag	The tag value; an integer from *0-4294967295*.
direct_rmap_name	The route-map name (string).
redist_bgp	Whether redistribute BGP is enabled; one of *Enable, Disable*.

Element	Description
`bgp_metric`	Redistribute BGP cost; an integer from *0-16777214.*
`bgp_metric_type`	The external metric type (integer); one of *1,2*.
`bgp_tag`	The BGP tag value; an integer from *0-4294967295*.
`bgp_rmap_name`	The BGP route map name (string).
`redist_static`	Whether redistribute static is enabled (string); one of *Enable, Disable*.
`static_metric`	Redistribute static cost; an integer from *0-16777214*.
`static_metric_type`	The external metric type (integer); one of *1,2*.
`static_tag`	The tag value; an integer from *0-4294967295*.
`static_rmap_name`	The static route map name (string).

Set OSPF Redistribute

Updates the OSPF redistribution.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/redistribute/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"redist_direct":"<*redist_direct*>",
	"direct_metric":"<*direct_metric*>",
	"direct_metric_type":"<*direct_metric_type*>",
	"direct_tag":"<*direct_tag*>",
	"direct_rmap_name":"<*direct_rmap_name*>",
	"redist_bgp":"<*redist_bgp*>",
	"bgp_metric":"<*bgp_metric*>",
	"bgp_metric_type": "<*bgp_metric*>",
	"bgp_tag":" "<*bgp_tag*>",
	"bgp_rmap_name":"<*bgp_rmap_name*>",
	"redist_static":"<redist_static>",
	"static_metric":"<*static_metric*>",
	"static_metric_type":"<*static_metric_type*>",
	"static_tag":"<*static_tag*>",
	"static_rmap_name":"<*static_rmap_name*>"
	}

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`redist_direct`	Redistribute the direct configuration (string); one of *Enable, Disable*.
`direct_metric`	Redistribute the direct cost; an integer from *0-16777214*.
`direct_metric_type`	The external metric type (integer); one of *1,2*.
`direct_tag`	The tag value; an integer from *0-4294967295*.
`direct_rmap_name`	The route-map name (string).
`redist_bgp`	Whether redistribute BGP is enabled; one of *Enable, Disable*.
`bgp_metric`	Redistribute BGP cost; an integer from *0-16777214*.
`bgp_metric_type`	The external metric type (integer); one of *1,2*.
`bgp_tag`	The BGP tag value; an integer from *0-4294967295*.
`bgp_rmap_name`	The BGP route map name (string).

Element	Description
redist_static	Whether redistribute static is enabled (string); one of *Enable, Disable*.
static_metric	Redistribute static cost; an integer from *0-16777214*.
static_metric_type	The external metric type (integer); one of *1,2*.
static_tag	The tag value; an integer from *0-4294967295*.
static_rmap_name	The static route map name (string).

Response

Response Body
 	
(JSON)
	{
	"vrfName": "*<vrfName>*",
	"redist_direct":"*<redist_direct>*",
	"direct_metric":"*<direct_metric>*",
	"direct_metric_type":"*<direct_metric_type>*",
	"direct_tag":"*<direct_tag>*",
	"direct_rmap_name":"*<direct_rmap_name>*",
	"redist_bgp":"*<redist_bgp>*",
	"bgp_metric":"*<bgp_metric>*",
	"bgp_metric_type": "*<bgp_metric>*",
	"bgp_tag":" "*<bgp_tag>*",
	"bgp_rmap_name":"*<bgp_rmap_name>*",
	"redist_static":"<redist_static>",
	"static_metric":"*<static_metric>*",
	"static_metric_type":"*<static_metric_type>*",
	"static_tag":"*<static_tag>*",
	"static_rmap_name":"*<static_rmap_name>*"
	}

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: *default*.
redist_direct	Redistribute the direct configuration (string); one of *Enable, Disable*.
direct_metric	Redistribute the direct cost; an integer from *0-16777214*.
direct_metric_type	The external metric type (integer); one of *1,2*.
direct_tag	The tag value; an integer from *0-4294967295*.
direct_rmap_name	The route-map name (string).
redist_bgp	Whether redistribute BGP is enabled; one of *Enable, Disable*.
bgp_metric	Redistribute BGP cost; an integer from *0-16777214*.

Element	Description
`bgp_metric_type`	The external metric type (integer); one of *1,2*.
`bgp_tag`	The BGP tag value; an integer from *0-4294967295*.
`bgp_rmap_name`	The BGP route map name (string).
`redist_static`	Whether redistribute static is enabled (string); one of *Enable, Disable*.
`static_metric`	Redistribute static cost; an integer from *0-16777214*.
`static_metric_type`	The external metric type (integer); one of *1,2*.
`static_tag`	The tag value; an integer from *0-4294967295*.
`static_rmap_name`	The static route map name (string).

Get OSPF NSSA area

Gets the OSPF nssa area configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/ospf/nssa/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.

Response

Response Body
 	
(JSON)
	{
	"nssa_area":"<*nssa_area*>",
	"nssa_area_id":"<*nssa_area_id*>",
	"nssa_def_info":"<*nssa_def_info*>",
	"nssa_def_metric":"<*nssa_def_metric*>",
	"nssa_def_metric_type":"<*nssa_def_metric_type*>",
	"nssa_no_redist":"<*nssa_no_redist*>",
	"nssa_no_summary":"<*nssa_no_summary*>",
	"nssa_translate_always": "<*nssa_translate_always*>",
	"nssa_stability_interval": "<*nssa_stability_interval*>"
	}

where:

Element	Description
`nssa_area`	The NSSA area configuration (string); one of *Enable, Disable*.
`nssa_area_id`	The NSSA area ID IP address (string); a valid IP address.
`nssa_def_info`	The NSSA default information originate configuration (string); one of *Enable, Disable*.
`nssa_def_metric`	The NSSA default metric; an integer from *0-16777214*.
`nssa_def_metric_type`	The NSSA external metric type (integer); one of *1,2*.
`nssa_no_redist`	Whether to stop redistribution in the NSSA area (string); one of *Enable, Disable*.

Element	Description
`nssa_no_summary`	Whether to stop summary LSAs into the NSSA area (string); one of *Enable, Disable*.
`nssa_translate_always`	`Always translate type7 LSA` (string); one of *Enable, Disable*.
`nssa_stability_interval`	The NSSA stability interval; an integer from *0-2147483647*.

Set OSPF NSSA area

Updates the OSPF nssa area configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/nssa/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"vrfName": "<*vrfName*>"
	"nssa_area":"<*nssa_area*>",
	"nssa_area_id":"<*nssa_area_id*>",
	"nssa_def_info":"<*nssa_def_info*>",
	"nssa_def_metric":"<*nssa_def_metric*>",
	"nssa_def_metric_type":"<*nssa_def_metric_type*>",
	"nssa_no_redist":"<*nssa_no_redist*>",
	"nssa_no_summary":"<*nssa_no_summary*>",
	"nssa_translate_always": "<*nssa_translate_always*>",
	"nssa_stability_interval": "<*nssa_stability_interval*>"
	}

where:

Element	Description
`vrf_name`	(Optional) Default VRF name. Default value: `default`.
`nssa_area`	The NSSA area configuration (string); one of *Enable, Disable*.
`nssa_area_id`	The NSSA area ID IP address (string); a valid IP address.
`nssa_def_info`	The NSSA default information originate configuration (string); one of *Enable, Disable*.
`nssa_def_metric`	The NSSA default metric; an integer from *0-16777214*.
`nssa_def_metric_type`	The NSSA external metric type (integer); one of *1,2*.
`nssa_no_redist`	Whether to stop redistribution in the NSSA area (string); one of *Enable, Disable*.
`nssa_no_summary`	Whether to stop summary LSAs into the NSSA area (string); one of *Enable, Disable*.
`nssa_translate_always`	`Always translate type7 LSA` (string); one of *Enable, Disable*.
`nssa_stability_interval`	The NSSA stability interval; an integer from *0-2147483647*.

Response

Response Body
 	
(JSON)
	{
	"nssa_area":"*<nssa_area>*",
	"nssa_area_id":"*<nssa_area_id>*",
	"nssa_def_info":"*<nssa_def_info>*",
	"nssa_def_metric":"*<nssa_def_metric>*",
	"nssa_def_metric_type":"*<nssa_def_metric_type>*",
	"nssa_no_redist":"*<nssa_no_redist>*",
	"nssa_no_summary":"*<nssa_no_summary>*",
	"nssa_translate_always": "*<nssa_translate_always>*",
	"nssa_stability_interval": "*<nssa_stability_interval>*"
	}

where:

Element	Description
`nssa_area`	The NSSA area configuration (string); one of *Enable, Disable*.
`nssa_area_id`	The NSSA area ID IP address (string); a valid IP address.
`nssa_def_info`	The NSSA default information originate configuration (string); one of *Enable, Disable*.
`nssa_def_metric`	The NSSA default metric; an integer from *0-16777214*.
`nssa_def_metric_type`	The NSSA external metric type (integer); one of *1,2*.
`nssa_no_redist`	Whether to stop redistribution in the NSSA area (string); one of *Enable, Disable*.
`nssa_no_summary`	Whether to stop summary LSAs into the NSSA area (string); one of *Enable, Disable*.
`nssa_translate_always`	`Always translate type7 LSA` (string); one of *Enable, Disable*.
`nssa_stability_interval`	The NSSA stability interval; an integer from *0-2147483647*.

Set OSPF default cost

Gets the OSPF database.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/area_def_cost

Request Body
 	
(JSON)
	{
	"vrfName": "*<vrfName>*",
	"area_id":"*<area_id>*",
	"state":"*<state>*",
	"default-cost":"*<default-cost>*"
	}

where:

Element	Description
`vrf_name`	(Optional) Default VRF name. Default value: *default*.
`area_id`	(Mandatory) The area ID IP address (string); a valid IP address.
`state`	Whether the default cost is enabled (string); one of *Enable, Disable*.
`default_cost`	The default summary cost value; an integer from *0-16777214*.

Response

Response Body (JSON)	

Set Area Authentication

Updates the OSPF area authentication.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/area_auth/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"vrfName": "<*vrfName*>",
	"area_id":"<*area_id*>",
	"auth":"auth",
	"message-digest":"<*message-digest*>"
	}

where:

Element	Description
`vrf_name`	(Optional) Default VRF name. Default value: *default*.
`area_id`	The area ID IP address (string); a valid IP address.
`auth`	The authentication configuration (string); one of *Enable, Disable*.

Response

Response Body (JSON)	

Set OSPF Summary Address

Gets the OSPF summary address.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/summary_addr/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"summary_addr": "<*summary_addr*>",
	"prefix":"<*prefix*>",
	"masklen":"<*masklen*>",
	"not-advertise":"<*not-advertise*>",
	"tag":"<*tag*>"
	}

where:

Element	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`summary_addr`	Whether to enable summary address configuration (string); one of *Enable, Disable*.
`prefix`	The IP address (string); a valid IP address.
`masklen`	The mask length; an integer from *0-32*.
`not-advertise`	Whether to suppress routes that match the prefix (string); one of *Enable, Disable*.
`tag`	The tag value; an integer from *0-4294967295*.

Response

Response Body (JSON)	

Set OSPF Area Range

Sets the OSPF area range addresses.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/range/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"area_id": "<*area_id*>",
	"range":"<*range*>",
	"prefix":"<*prefix*>",
	"masklen":"<*masklen*>",
	"not-advertise":"<*not-advertise*>"
	}

where:

Variable	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`area_id`	The area ID IP address (string); a valid IP address.
`range`	Whether to enable the range of IP addresses (string); one of *Enable, Disable*.
`prefix`	The IP address (string); a valid IP address.
`mask`	The mask length; an integer from *0-32*.
`not-advertise`	Whether to suppress routes that match the prefix (string); one of *Enable, Disable*.

Response

Response Body (JSON)	

Set OSPF Overflow Database

Updates the database overflow limits.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/overflow_db/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"db_overflow": "<*db_overflow*>",
	"max_lsas":"<*max_lsas*>",
	"limit":"<*limit*>",
	"external":"<*external*>",
	"ext_max_lsas":"<*ext_max_lsas*>",
	"recovery_time":"<*recovery_time*>"
	}

where:

Variable	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`db_overflow`	Whether to enable the database overflow configuration (string); one of *Enable, Disable*.
`max_lsas`	The maximum LSA limit; an integer from *0-4294967294*.
`limit`	The database limit type (string); one of *Hard, soft*.
`external`	Whether to enable the external LSA limit (string); one of *Enable, Disable*.
`ext_max_lsas`	The maximum external LSA limit; an integer from *0-2147483647*.
`recovery_time`	Tme time to recover from the external LSA limit, in seconds; an integer from *0-65535*.

Response

Response Body (JSON)	

Set OSPF Auto-cost Reference Bandwidth

Updates the auto-cost reference bandwidth.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/autocost_refbw/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"autocost_refbw": "<*autocost_refbw*>",
	"bw_value":"<*bw_value*>",
	"bw_unit":"<*bw_unit*>"
	}

where:

Variable	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`autocost_refbw`	Whether to enable the autocost reference bandwidth configuration (string); one of *Enable, Disable*.
`bw_value`	The bandwidth value; an integer from: • *1-4294* for Gbps • *1-4294967* for Mbps
`bw_unit`	The bandwidth unit (string); one of *Gbps, Mbps*.

Response

Response Body (JSON)	

Set OSPF Stub Configuration

Updates the stub area configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/stub/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"area_id": "<*area_id*>",
	"stub":"<*stub*>",
	"no_summary":"<*no_summary*>"
	}

where::

Variable	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`area-id`	The area ID IP address (string); a valid IP address.
`stub`	Whether to enable the stub area configuration (string); one of *Enable, Disable*.
`no_summary`	Whether to not inject summary routes into the stub configuration (string); one of *Enable, Disable*.

Response

Response Body (JSON)	

Set OSPF Remove Configuration

Sets the OSPF remove commands.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ospf/clear/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"process": "<*process*>",
	"statistics":"<*statistics*>",
	"traffic":"<*traffic*>",
	"neighbors":"<*neighbors*>"
	}

where:

Variable	Description
vrf_name	(Optional) Default VRF name. Default value: `default`.
`process`	Remove the OSPF process configurations (string); one of *Enable, Disable*.
`statistics`	Remove the OSPF statistic configurations (string); one of *Enable, Disable*.
`traffic`	Remove the OSPF process traffic statistic configurations (string); one of *Enable, Disable*.
`neighbors`	Remove the OSPF neighbor configurations (string); one of *Enable, Disable*.

Response

Response Body (JSON)	

PKA

The following Public Key Authentication (PKA) URIs are available:

- /nos/api/cfg/pka GET
- /nos/api/cfg/pka/pub GET, PUT,DELETE

The following PKA commands are available:

- Get PKA Summary Informations
- Get PKA Certificate
- Import PKA Certificate via SFTP
- Import PKA Certificate Directly
- Delete PKA Certificate

Get PKA Summary Informations

Gets the PKA summary informations.

Request

Method Type	GET
Request URI	/nos/api/cfg/pka
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>[
	{
	"pka_number": {pka_number},
	"label_name_list": {label_name_list},
	“pka_username“: {pka_username}
	}
	]</pre>

where:

Element	Description
`pka_number`	The binding public key number for a specific user; an integer from *1-10*.
`label_name_list`	The binding public key label name list for a specific user a string in the following format: *{label_name_list}*.
`pka_username`	The username who binds the public key; an integer up to 28 characters.

Get PKA Certificate

Gets the public key certificate for a specific label name.

Request

Method Type	GET
Request URI	/nos/api/cfg/pka/pub/<*pka_username*>/<*label_name*>
Request Body (JSON)	

where:

Element	Description
pka_username	(Mandatory) The username who binds the public key; an integer up to 28 characters.
label_name	(Mandatory) The binding public key label name for a specific user a string in the following format: *{label_name}.*

Response

Response Body (JSON)	["ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDZXSHylc4OU9ByMtHoC2E9K1Onpyo tacOMcTKP/zAXRbGeZT9CU58LPLneRYzkZQ1o6EQsOHx+Ocodt6kYfOnqVY sl5xRrKPNQYSxUVQYBwKZCigb7LwUPogaiX81hOl2OsMxAzbLTx3YydzEtl SELfzPdqj+FHErycMUyOmmc1azIJc/USl/ZBmvW7KOUdVjwj1DBcQlZYdUX YNxKG/+YR3LpdpHkJnsxtDobdW94G3rqR2bTdcHXWcrZjCnpzQEcYjrDwHS dO9EJwQZ5a+KoRtkuZsYYyqP5s/jAwyk4+B5saRidtD2n4H3qzKCq7U4PpZ EiFF3DOsgcU/ODu7fT stack@ubuntu-226"]

Import PKA Certificate via SFTP

Imports the public key certificate for a specific label name.

Request

Method Type	PUT
Request URI	/nos/api/cfg/pka/pub/<*pka_username*>/<*label_name*>

Request Body
 	
(JSON)
	<pre>{
	  "import_type":"sftp",
	  "server_ip":{server_ip},
	  "src_file":{src_file},
	  "login_username":{login_username},
	  "login_passwd":{login_password},
	  "vrf_name":"{management | default}"
	}</pre>

where:

Element	Description
pka_username	(Mandatory) The username who binds the public key; an integer up to 28 characters.
label_name	(Mandatory) The binding public key label name for a specific user a string in the following format: `{label_name}`.
`import_type`	(Mandatory) The import type for the public key certificate (string); one of *sftp,line*.
`server_ip`	(Mandatory) The server IP address; a valid IPv4 or IPv6 address.
`src_file`	(Mandatory) The source file; a string up to 256 characters.
`login_username`	(Mandatory) The server username (string).
`login_passwd`	(Mandatory) The server password (string).
`vrf_name`	(Optional) The VRF instance name. Valid values: • *default* • *management* Default value: *default*.

Response

Response Body
 	
(JSON)
	<pre>{
	    "status": "{successful | failed}",
	    "details": "{details}",
	    "filename": "{ filename }"
	}</pre>

where:

Element	Description
`status`	The import status (string); one of *failed,import success*.
`details`	The detailed description of the status (string).
`filename`	The source file name (string).

Import PKA Certificate Directly

Imports the public key content.

Request

Method Type	PUT
Request URI	/nos/api/cfg/pka/pub/<*pka_username*>/<*label_name*>

Request Body
 	
(JSON)
	{
	  "import_type":"line",
	  "pka_content": {pka_content}
	}

where:

Element	Description
pka_username	The username who binds the public key; an integer up to 28 characters.
label_name	The binding public key label name for a specific user a string in the following format: `{label_name}`.
`import_type`	The import type for the public key certificate (string); one of *sftp,line*.
`pka_content`	The public key certificate content; a string in the following format: *{pka_content}*.

Response

Response Body
 	
(JSON)
	{
	  "status": "{successful | failed}",
	  "details": "{details}",
	  "filename": "{filename}"
	}

where:

Element	Description
`status`	The import status (string); one of *failed,import success*.
`details`	The detailed description of the status (string).
`filename`	The source file name (string). If the import type is *line*, the filename is *null*.

Delete PKA Certificate

Deletes a public key certificate.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pka/pub/<*pka_username*>/<*label_name*>
Request Body (JSON)	

where:

Element	Description
pka_username	(Mandatory) The username who binds the public key; an integer up to 28 characters.
label_name	(Mandatory) The binding public key label name for a specific user a string in the following format: *{label_name}*.

Response

Response Body (JSON)	

PKI

The following Private Key Infrastructure (PKI) URIs are available:

- /nos/api/cfg/pki GET, DELETE
- /nos/api/cfg/pki/ca GET, PUT, DELETE
- /nos/api/cfg/pki/hostcert GET, PUT, POST, DELETE
- /nos/api/cfg/pki/csr GET, PUT, POST

The following PKI commands are available:

- Get PKI Profile Summary Information
- Delete PKI Profile
- Get CA Certificate
- Import CA Certificate
- Delete CA Certificate
- Get Host Certificate
- Import Host Certificate
- Generate Host Certificate
- Delete Host Certificate
- Generate CSR
- Get CSR
- Export CSR and Import Signed CSR

Get PKI Profile Summary Information

Gets summary information about a specific PKI profile or about all configured PKI profiles.

Request

Method Type	GET
Request URI	/nos/api/cfg/pki/<*pki_name*>
Request Body (JSON)	

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long. **Note:** If no PKI profile name is provided, then the command returns summary information about all configured PKI profiles.

Response

Response Body (JSON)	[
	{
	"PKI_Profile_Name":"<*pki_name*>",
	"In_use":"{yes|no}",
	"CA":"<*ca_number*>",
	"Host_certificate":"{existed|non-existed}",
	"CSR":"{existed|non-existed}"
	}
	]

where:

Element	Description
`PKI_Profile_Name`	The name of the PKI profile; a string up to 16 characters long.
`In_use`	Whether the PKI profile is used by the switch; one of *yes, no*.
`CA`	The number of Certificate Authority (CA) stored in the PKI profile; an integer from *0-5*.
`Host_certificate`	Whether host certificates are present in the PKI profile; one of *existed, non-existed*.
`CSR`	Whether Certificate Signing Requests (CSR) are present in the PKI profile; one of *existed, non-existed*.

Delete PKI Profile

Removes a configured PKI profile.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pki/<*pki_name*>
Request Body (JSON)	

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long.

Get CA Certificate

Gets the Certificate Authority (CA) stored in the PKI profile.

Request

Method Type	GET
Request URI	/nos/api/cfg/pki/ca/<*pki_name*>/<*format*>
Request Body (JSON)	

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long.
format	The format for displaying the CA; one of `text, base64`. Default value: `text`.

Response

Example when chosen format is *text*:

Response Body (JSON)

```
[
 "Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number:  89:0f:d1:1e:d7:79:37:cb
        Signature Algorithm: sha256WithRSAEncryption
        Issuer: C=US, ST=Maryland, L=Baltimore, O=Test CA,
Limited, OU=Server Research Department, CN=Test CA,
emailAddress=test@example.com
        Validity
            Not Before: Dec 28 08:26:50 2016 GMT
            Not After : Sep 24 08:26:50 2019 GMT
        Subject: C=US, ST=Maryland, L=Baltimore, O=Test CA,
Limited, OU=Server Research Department, CN=Test
CA/emailAddress=test@example.com
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
                Public-Key: (4096 bit)
                Modulus:
00:a8:cc:51:e8:18:c7:50:68:12:3c:7d:c4:9f:9a:
d3:a3:4e:8d:0a:e3:b6:25:88:f6:07:0e:de:09:b3:
96:59:d7:e3:38:2c:2f:1d:50:e5:40:91:4f:29:9d:
b6:45:4f:d6:f3:f9:27:09:04:85:13:50:91:6b:e8:
4e:6b:c0:f7:c5:b9:f0:9b:45:9a:90:80:49:5f:3b:
...
```

Example when chosen format is *base64*:

Response Body (JSON)	["-----BEGIN CERTIFICATE----- MIIGMzCCBBugAwIBAgIJAIkPOR7XeTfLMAOGCSqGSIb3DQEBCwUAMIGnMQswCQYD VQQGEwJVUzERMA8GA1UECAwITWFyeWxhbmQxEjAQBgNVBAcMCUJhbHRpbW9yZTEZ MBcGA1UECgwQVGVzdCBDQSwgTGltaXRlZDEjMCEGA1UECwwaU2VydmVyIFJlc2Vh cmNoIERlcGFydG1lbnQxEDAOBgNVBAMMB1Rlc3QgQOExHzAdBgkqhkiG9wOBCQEW EHRlc3RAZXhhbXBsZS5jb2OwHhcNMTYxMjI4MDgyNjUwWhcNMTkwOTIOMDgyNjUw WjCBpzELMAkGA1UEBhMCVVMxETAPBgNVBAgMCE1hcnlsYW5kMRIwEAYDVQQHDAlC YWxOaW1vcmUxGTAXBgNVBAoMEFRlc3QgQOEsIExpbWlOZWQxIzAhBgNVBAsMGlNl cnZlciBSZXNlYXJjaCBEZXBhcnRtZW5OMRAwDgYDVQQDDAdUZXNOIENBMR8wHQYJ KoZIhvcNAQkBFhBOZXNOQGV4YW1wbGUuY29tMIICIjANBgkqhkiG9wOBAQEFAAOC Ag8AMIICCgKCAgEAqMxR6BjHUGgSPH3En5rToO6NCuO2JYj2Bw7eCbOWWdfjOCwv HVDlQJFPKZ22RU/W8/knCQSFE1CRa+hOa8D3xbnwmOWakIBJXzuLscAIIV78nDuq NzI6xbuHdTOP6OldZiLUgyqZypbOScNfGkQnCMoSIdJBUpAgkqG9hGYD/ATpn3/R BzdrOGCPJZ3Zm/9Oi5qUBKHoQmXdk5/R29kf4bM2HLKW2tqh4+Ba/PJoWZ2wASvT AzU39JTkLU+HtKg8F2hEs4CtZ/sE1A+oKSGUzfTJPf5OmM28xIb4aOHlecgTY92i /6rbcdxZS1ORnH3ZlVENlHgQCeYf4XXBLI2OkL1Bknm5IqqojKlBNP7Z61vEiLhx LqX8XdXyrjzqXK7PO+yc+7jXmwG9mhPxWDWO5yvFFMlIhwMNhO8uNE+dCo+REmtp jGkmi4gJdzUEDZwYfUXbIRZpdAtCYuIygZOCX7kzvSoGn2IxG43svKhxZYvRO38M xtjjBNW23jMcpMy+48bsO4dVMBNYJd9LUbI1+tMKqZZmTT8dcRP6woxLO5DCVn79 ... Ay8jxzXX93uIKaOa3K+OB25iFT6xwK5OioVSLfaGqwfw7lsf/k2vFViqhBTnoLux 2oSiHoEHGGSAMwOpQz9DTdVFYeFieE+BaIfsOyEgmXoJC4sxUrDrcpzJydSfjJ6n 5Pc2cZRVkJZoSWXR4oNfJCsvpxTDJrN8A+3UqyIW/539shRBqUyxounE25ooYN49 aORIjMDXm25bzf7V7Xm+uHUw7gQeWACLIohJiqc7QS+Ow1b9zBau5pU67ONH+W7R LjE17rXskA== -----END CERTIFICATE-----"]

Import CA Certificate

Imports a CA certificate from a remote server to a specific PKI profile on the switch.

Request

Method Type	PUT
Request URI	/nos/api/cfg/pki/ca/<*pki_name*>

Request Body (JSON)	{
	"server_ip":"{*server_ip*}",
	"src_file":"{*src_file*}",
	"username":"{*username*}",
	"passwd":"{*password*}",
	"vrf_name":"{management|default}"
	}

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long. **Note:** If the specified PKI profile does not exist, then the switch creates a new PKI profile with that name.
`server_ip`	The IP address of the remote server from where the CA certificate is imported.
`src_file`	The source filepath of the CA certificate on the server; a string representing the filepath of the CA certificate.
`username`	The username used to log onto the remote server (string).
`passwd`	The password used to log onto the remote server (string).
`vrf_name`	The VRF instance used to connect to the remote server; one of *default, management.* Default value: *default.*

Response

Response Body (JSON)	{
	"status":"{successful|failed}",
	"details":"{*details*}",
	"filename":"{*filename*}"
	}

where:

Element	Description
`status`	The status of the CA certificate import; one of *successful, failed.*
`details`	The detailed description of the import status (string).
`filename`	The filename of the CA certificate (string).

Delete CA Certificate

Deletes a CA certificate from a specific PKI profile.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pki/ca/<*pki_name*>?subject=<*subject*>
Request Body (JSON)	

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long.
subject	The subject of the CA certificate; a string up to 255 characters long. For example: "C=US, ST=Maryland, L=Baltimore, O=Test CA, Limited, OU=Server Research Department, CN=Test CA, emailAddress=test@example.com"

Get Host Certificate

Gets the host certificate stored in a specific PKI profile.

Request

Method Type	GET
Request URI	/nos/api/cfg/pki/hostcert/<*pki_name*>/<*format*>
Request Body (JSON)	

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long.
format	The format for displaying the host certificate; one of `text`, `base64`. Default value: `text`.

Response

Example when chosen format is *text*:

Response Body (JSON)

```
[
    "Certificate:
      Data:
      Version: 3 (0x2)
      Serial Number: 89:0f:d1:1e:d7:79:37:cb
      Signature Algorithm: sha256WithRSAEncryption
      Issuer: C=US, ST=Maryland, L=Baltimore, O=Test CA,
Limited, OU=Server Research Department, CN=Test CA,
emailAddress=test@example.com
      Validity
         Not Before: Dec 28 08:26:50 2016 GMT
         Not After : Sep 24 08:26:50 2019 GMT
      Subject: C=US, ST=Maryland, L=Baltimore, O=Test CA,
Limited, OU=Server Research Department, CN=Test
CA/emailAddress=test@example.com
      Subject Public Key Info:
         Public Key Algorithm: rsaEncryption
            Public-Key: (4096 bit)
            Modulus:
00:a8:cc:51:e8:18:c7:50:68:12:3c:7d:c4:9f:9a:
d3:a3:4e:8d:0a:e3:b6:25:88:f6:07:0e:de:09:b3:
96:59:d7:e3:38:2c:2f:1d:50:e5:40:91:4f:29:9d:
b6:45:4f:d6:f3:f9:27:09:04:85:13:50:91:6b:e8:
4e:6b:c0:f7:c5:b9:f0:9b:45:9a:90:80:49:5f:3b:
...
```

Example when chosen format is *base64*:

Response Body (JSON)	["-----BEGIN CERTIFICATE----- MIIGMzCCBBugAwIBAgIJAIkPOR7XeTfLMAOGCSqGSIb3DQEBCwUAMIGnMQswCQYD VQQGEwJVUzERMA8GA1UECAwITWFyeWxhbmQxEjAQBgNVBAcMCUJhbHRpbW9yZTEZ MBcGA1UECgwQVGVzdCBDQSwgTGltaXRlZDEjMCEGA1UECwwaU2VydmVyIFJlc2Vh cmNoIERlcGFydG1lbnQxEDAOBgNVBAMMB1Rlc3QgQOExHzAdBgkqhkiG9wOBCQEW EHRlc3RAZXhhbXBsZS5jb2OwHhcNMTYxMjI4MDgyNjUwWhcNMTkwOTIOMDgyNjUw WjCBpzELMAkGA1UEBhMCVVMxETAPBgNVBAgMCE1hcnlsYW5kMRIwEAYDVQQHDAlC YWxOaW1vcmUxGTAXBgNVBAoMEFRlc3QgQOEsIExpbWlOZWQxIzAhBgNVBAsMGlNl cnZlciBSZXNlYXJjaCBEZXBhcnRtZW5OMRAwDgYDVQQDDAdUZXNOIENBMR8wHQYJ KoZIhvcNAQkBFhBOZXNOQGV4YW1wbGUuY29tMIICIjANBgkqhkiG9wOBAQEFAAOC Ag8AMIICCgKCAgEAqMxR6BjHUGgSPH3En5rToO6NCuO2JYj2Bw7eCbOWWdfjOCwv HVDlQJFPKZ22RU/W8/knCQSFE1CRa+hOa8D3xbnwmOWakIBJXzuLscAIIV78nDuq NzI6xbuHdTOP6OldZiLUgyqZypbOScNfGkQnCMoSIdJBUpAgkqG9hGYD/ATpn3/R BzdrOGCPJZ3Zm/9Oi5qUBKHoQmXdk5/R29kf4bM2HLKW2tqh4+Ba/PJoWZ2wASvT AzU39JTkLU+HtKg8F2hEs4CtZ/sE1A+oKSGUzfTJPf5OmM28xIb4aOHlecgTY92i /6rbcdxZS1ORnH3ZlVENlHgQCeYf4XXBLI2OkL1Bknm5IqqojKlBNP7Z61vEiLhx LqX8XdXyrjzqXK7PO+yc+7jXmwG9mhPxWDWO5yvFFMlIhwMNhO8uNE+dCo+REmtp jGkmi4gJdzUEDZwYfUXbIRZpdAtCYuIygZOCX7kzvSoGn2IxG43svKhxZYvRO38M xtjjBNW23jMcpMy+48bsO4dVMBNYJd9LUbI1+tMKqZZmTT8dcRP6woxLO5DCVn79 JCAqCVYOvFE1mVSUggF5CVGOn9/dylTmXIpaiapOdAqJ3DNjnPbxpCnUYXn3NFIL ... Ay8jxzXX93uIKaOa3K+OB25iFT6xwK5OioVSLfaGqwfw7lsf/k2vFViqhBTnoLux 2oSiHoEHGGSAMwOpQz9DTdVFYeFieE+BaIfsOyEgmXoJC4sxUrDrcpzJydSfjJ6n 5Pc2cZRVkJZoSWXR4oNfJCsvpxTDJrN8A+3UqyIW/539shRBqUyxounE25ooYN49 aORIjMDXm25bzf7V7Xm+uHUw7gQeWACLIohJiqc7QS+Ow1b9zBau5pU67ONH+W7R LjE17rXskA== -----END CERTIFICATE-----"]

Import Host Certificate

Imports a host certificate from a remote server to a specific PKI profile on the switch.

Request

Method Type	PUT
Request URI	/nos/api/cfg/pki/hostcert/<*pki_name*>

Request Body (JSON)	{
	"server_ip":"{*server_ip*}",
	"src_file":"{*src_file*}",
	"dst_file":"{certificate|private-key}",
	"username":"{*username*}",
	"passwd":"{*password*}",
	"vrf_name":"{management|default}"
	}

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long. **Note:** If the specified PKI profile does not exist, then the switch creates a new PKI profile with that name.
`server_ip`	The IP address of the remote server from where the host certificate is imported.
`src_file`	The source filepath of the host certificate on the server; a string representing the filepath of the CA certificate.
`dst_file`	The destination filepath of the host certificate on the switch; one of *certificate, private-key.* **Note:** The host certificate is imported only when both the certificate and the private key are successfully imported.
`username`	The username used to log onto the remote server (string).
`passwd`	The password used to log onto the remote server (string).
`vrf_name`	The VRF instance used to connect to the remote server; one of *default, management.* Default value: *default.*

Response

Response Body (JSON)	{
	"status":"{successful|failed}",
	"details":"{*details*}",
	"filename":"{*filename*}"
	}

where:

Element	Description
`status`	The status of the host certificate import; one of *successful, failed.*
`details`	The detailed description of the import status (string). If only the certificate has been imported, then this variable is "*still needs private-key*". If only the private key has been imported, the this variable is "*still needs certificate*".
`filename`	The filename of the host certificate (string).

Generate Host Certificate

Generates a host certificate for a specific PKI profile.

Request

Method Type	POST
Request URI	/nos/api/cfg/pki/hostcert/<*pki_name*>

Request Body (JSON)	{
	"CountryName":"{*CountryName*}",
	"StateName":"{*StateName*}",
	"LocalityName":"{*LocalityName*}",
	"OrganizationName":"{*OrganizationName*}",
	"OrganizationalUnitName":"{*OrganizationalUnitName*}",
	"CommonName":"{*CommonName*}",
	"Email":"{*Email*}"
	}

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long. **Note:** If the specified PKI profile does not exist, then the switch creates a new PKI profile with that name.
`CountryName`	The code of the country; a string up to 2 characters long. For example: US.
`StateName`	The name of the state or province; a string up to 16 characters long. For example: California.
`LocalityName`	The name of the locality (for example, a city); a string up to 32 characters long. For example: Santa Clara.
`OrganizationName`	The name of the organization (for example, a company); a string up to 64 characters long. For example: Lenovo.
`OrganizationalUnitName`	The name of the organizational unit (for example, a department); a string up to 32 characters long. For example: Network Engineering.
`CommonName`	The common name; a string up to 64 characters long. For example: TestHostCert. **Note:** The name must start with an alphanumeric character and can contain only alphanumeric characters and the underscore (_), dash (-) and point (.) characters.
`Email`	The e-mail address; a string up to 32 characters long. For example: test@example.com.

Response

Response Body (JSON)	{
	"status":"{generation succeeded|generation failed}",
	}

where:

Element	Description
`status`	The status of the host certificate generation; one of *generation succeeded, generation failed.*

Delete Host Certificate

Deletes the host certificate from a specific PKI profile.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pki/hostcert/<*pki_name*>
Request Body (JSON)	

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long.

Generate CSR

Generates the Certificate Signing Request (CSR) for a specific PKI profile.

Request

Method Type	POST
Request URI	/nos/api/cfg/pki/csr/<*pki_name*>

Request Body (JSON)	{
	"CountryName":"{*CountryName*}",
	"StateName":"{*StateName*}",
	"LocalityName":"{*LocalityName*}",
	"OrganizationName":"{*OrganizationName*}",
	"OrganizationalUnitName":"{*OrganizationalUnitName*}",
	"CommonName":"{*CommonName*}",
	"Email":"{*Email*}"
	}

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long. **Note:** If the specified PKI profile does not exist, then the switch creates a new PKI profile with that name.
`CountryName`	The code of the country; a string up to 2 characters long. For example: US.
`StateName`	The name of the state or province; a string up to 16 characters long. For example: California.
`LocalityName`	The name of the locality (for example, a city); a string up to 32 characters long. For example: Santa Clara.
`OrganizationName`	The name of the organization (for example, a company); a string up to 64 characters long. For example: Lenovo.
`OrganizationalUnitName`	The name of the organizational unit (for example, a department); a string up to 32 characters long. For example: Network Engineering.
`CommonName`	The common name; a string up to 64 characters long. For example: TestCSR. **Note:** The name must start with an alphanumeric character and can contain only alphanumeric characters and the underscore (_), dash (-) and point (.) characters.
`Email`	The e-mail address; a string up to 32 characters long. For example: test@example.com.

Response

Response Body (JSON)	{
	"status":"{generation succeeded|generation failed}",
	}

where:

Element	Description
`status`	The status of the host certificate generation; one of *generation succeeded, generation failed.*

Get CSR

Gets the CSR stored in a specific PKI profile.

Request

Method Type	GET
Request URI	/nos/api/cfg/pki/csr/<*pki_name*>/<*format*>
Request Body (JSON)	

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long.
format	The format for displaying the CSR; one of `text`, `base64`. Default value: `text`.

Response

Example when chosen format is *text*:

Response Body (JSON)

```
[
    "Certificate Request:
    Data:
        Version: 0 (0x0)
        Subject: C=US, ST=California, L=Santa Clara,
O=Lenovo Networking Operating System, OU=Network
Engineering, CN=cccc/emailAddress=cccc@lenovo.com
        Subject Public Key Info:
           Public Key Algorithm: rsaEncryption
                Public-Key: (2048 bit)
                Modulus:
00:c3:78:73:d2:6b:e6:64:3c:10:8a:63:3d:11:d0:
bf:95:56:0b:a1:1c:12:39:6d:d8:5a:45:20:06:90:
b4:a9:2f:a6:87:5c:bc:fd:d3:1f:08:e6:31:7c:82:
94:aa:50:20:14:5a:63:19:1e:26:52:b0:35:56:b2:
8c:b6:aa:e1:6e:1c:1b:32:79:08:bd:02:a6:1b:5d:
59:cc:f1:44:12:36:e3:7d:2b:73:7e:5e:70:9e:e2:
b9:94:3c:38:5b:fd:30:10:2b:b7:7b:5d:d0:d9:08:
...
```

Example when chosen format is *base64*:

Response Body (JSON)	["-----BEGIN CERTIFICATE REQUEST----- MIIC+DCCAeACAQAwgbIxCzAJBgNVBAYTAlVTMRMwEQYDVQQIDApDYWxpZm9ybml h MRQwEgYDVQQHDAtTYW5OYSBDbGFyYTErMCkGA1UECgwiTGVub3ZvIE5ldHdvcmt p bmcgT3BlcmFOaW5nIFN5c3RlbTEcMBoGA1UECwwTTmVOd29yayBFbmdpbmVlcml u ZzENMAsGA1UEAwwEY2NjYzEeMBwGCSqGSIb3DQEJARYPY2NjYOBsZW5vdm8uY29 t MIIBIjANBgkqhkiG9wOBAQEFAAOCAQ8AMIIBCgKCAQEAw3hzOmvmZDwQimM9EdC / lVYLoRwSOW3YWkUgBpCOqS+mh1y8/dMfCOYxfIKUqlAgFFpjGR4mUrA1VrKMtqr h bhwbMnkIvQKmG11ZzPFEEjbjfStzfl5wnuK5lDw4W/OwECu3e13Q2QhKlCCu2gG x EQKXOMrC1cnnSEWp2OkoYSWMYAEGQkjyOuVNAEWxiMDb1EXf5yG2fwnGNMSZ+Of / qFQ+2cWIOWY/7GnyIrNWlWpZoD+oGAZBjnlqrMTR9bZTwAQMvgI99Ry9LeSXxaK 6 KH5xe14cSHM1KqNrv9aht3w+GalEQlEKUEgTwS4OwmMhYfSff81ZexRQNiDWVd2 j SQIDAQABoAAwDQYJKoZIhvcNAQELBQADggEBALTGf7fktJgKclLA6Zy8tVjOs2W U kwDGphm/cLV/bBovCTV/nFsmNP4o6qvyffdO8t3WAz9PIcfceg7bkJMOuSW8Ii7 c 5cKg59smpuNGNA9rnReB+M9Ocv3+Ah9SJQGYArbHImP8ApWyPMADREKwkGW8YsA G KqiGXsSs9xJ2ONAPCr/l8+6pvVcAn5L7iyCyNb69X73S8FTbXks+3VRzQv5VWuM z IwHIZjI6YniZVJjAZqpiqR9OEK4KrV4wzeUL2lJL/RLRRdRnRk3QfpJO9GxP+Hp e 1U4B2xvMIklLXIzUwHlaHqli34RscYW12moWh8/G125QqfsORE1YWXGPw24= -----END CERTIFICATE REQUEST-----"]

Export CSR and Import Signed CSR

Exports a CSR or imports a signed CSR.

Request

Method Type	PUT
Request URI	/nos/api/cfg/pki/csr/<*pki_name*>/<*action*>

Request Body (JSON)	{
	"server_ip":"{*server_ip*}",
	"src_file":"{*src_file*}",
	"username":"{*username*}",
	"passwd":"{*password*}",
	"vrf_name":"{management|default}"
	}

where:

Element	Description
pki_name	The name of the PKI profile; a string up to 16 characters long. **Note:** If the specified PKI profile does not exist, then the switch creates a new PKI profile with that name.
action	Whether to export a CSR or import a signed CSR; one of `export`, `import`.
`server_ip`	The IP address of the remote server from where the CA certificate is imported.
`src_file`	The source filepath of the CA certificate on the server; a string representing the filepath of the CA certificate
`username`	The username used to log onto the remote server (string).
`passwd`	The password used to log onto the remote server (string).
`vrf_name`	The VRF instance used to connect to the remote server; one of *default, management.* Default value: *default.*

Response

Response Body (JSON)	{
	"status":"{successful|failed}",
	"details":"{*details*}",
	"filename":"{*filename*}"
	}

where:

Element	Description
`status`	The status of the CA certificate import; one of *successful, failed.*

Element	Description
`details`	The detailed description of the import status (string).
`filename`	The filename of the CA certificate (string).

Private VLAN

The following Private VLAN-related URIs are available:

- /nos/api/cfg/pvlan GET, POST, DELETE
- /nos/api/cfg/pvlan/interface GET
- /nos/api/cfg/pvlan/<*pvlan_type*>/<*vid*> POST, DELETE
- /nos/api/cfg/pvlan/primary/ POST, DELETE
- /nos/api/cfg/pvlan/interface/<*if_type*> POST, PUT, DELETE
- nos/api/cfg/pvlan/primary/<*primary_pvlan_vid*> DELETE

The following PVLAN commands are available:

- Create a Private VLAN
- Delete a Private VLAN
- Create a Private VLAN Association
- Delete a Private VLAN Association
- Delete All Private VLAN Associations for a VLAN
- Apply a Private VLAN Mode on Interface
- Remove a Private VLAN Mode from Interface
- Create or Remove a Private VLAN Port Mapping/Association
- Show Private VLAN Information
- Show Interface Private VLAN Information
- Enable Private VLAN Globally
- Disable Private VLAN Globally

Create a Private VLAN

Creates a private vLAN.

Request

Method Type	POST
Request URI	/nos/api/cfg/pvlan/<*pvlan_type*>/<*vid*>
Request Body (JSON)	

where:

Element	Description
pvlan_type	(Mandatory) Private VLAN type; one of `primary`, `isolated`, `community`.
vid	(Mandatory) The VLAN ID; a positive integer from 2-4093.

Response

Response Body (JSON)	

Delete a Private VLAN

Deletes a private VLAN.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pvlan/<*pvlan_type*>/<*vid*>
Request Body (JSON)	

where:

Element	Description
pvlan_type	(Mandatory) Private VLAN type; one of `primary`, `isolated`, `community`.
vid	(Mandatory) The VLAN ID; a positive integer from 2-4093.

Response

Response Body (JSON)	

Create a Private VLAN Association

Creates a private VLAN association.

Request

Method Type	POST
Request URI	/nos/api/cfg/pvlan/primary/<*primary_pvlan_vid*>/assoc/<*secondary_vlan_id*>
Request Body (JSON)	

where:

Element	Description
primary_pvlan_vid	(Mandatory) The primary VLAN ID; a positive integer from 2-4093.
secondary_vlan_id	(Mandatory) The secondary VLAN ID; a positive integer from 2-4093.

Response

Response Body (JSON)	

Delete a Private VLAN Association

Deletes a private VLAN association.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pvlan/primary/<*primary_pvlan_vid*>/assoc/<*secondary_vlan_id*>
Request Body (JSON)	

where:

Element	Description
primary_pvlan_vid	(Mandatory) The primary VLAN ID; a positive integer from 2-4093.
secondary_vlan_id	(Mandatory) The secondary VLAN ID; a positive integer from 2-4093.

Response

Response Body (JSON)	

Delete All Private VLAN Associations for a VLAN

Deletes all private VLAN associations for a VLAN.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pvlan/primary/<*primary_pvlan_vid*>/assoc
Request Body (JSON)	

where:

Element	Description
primary_pvlan_vid	(Mandatory) The primary VLAN ID; a positive integer from 2-4093.

Response

Response Body (JSON)	

Apply a Private VLAN Mode on Interface

Applies a private VLAN mode on a specified interface.

Request

Method Type	POST
Request URI	/nos/api/cfg/pvlan/interface/<*if_type*>/<*chassis_number*>/<*if_number*>/<*if_sub_number*>
Request Body (JSON)	

where:

Element	Description
if_type	(Mandatory) The interface type (string); one of `ethernet`, for Ethernet interfaces, `po` for port-channel interfaces.
chassis_number	(Mandatory) The chassis number for the Ethernet interfaces (integer). For example: 1 for Ethernet1/2, or none for port-channel interfaces.
if_number	(Mandatory) The interface number for the Ethernet interfaces (integer). An integer from 1 to the maximum number of switch interfaces.
if_sub_number	(Optional) The sub-interface number for the Ethernet interfaces a positive integer from 1-4.

Response

Response Body (JSON)	

Remove a Private VLAN Mode from Interface

Deletes a private VLAN mode on a specified interface.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pvlan/interface/<*if_type*>/<*chassis_number*>/<*if_number*>/<*is_subnumber*>
Request Body (JSON)	

where:

Element	Description
if_type	(Mandatory) The interface type (string); one of `ethernet`, for Ethernet interfaces, `po` for port-channel interfaces.
chassis_number	(Mandatory) The chassis number for the Ethernet interfaces (integer). For example: 1 for Ethernet1/2, or none for port-channel interfaces.
if_number	(Mandatory) The interface number for the Ethernet interfaces (integer). An integer from 1 to the maximum number of switch interfaces.
if_sub_number	(Optional) The sub-interface number for the Ethernet interfaces a positive integer from 1-4.

Response

Response Body (JSON)	

Create or Remove a Private VLAN Port Mapping/Association

Creates or deletes a private VLAN port mapping/association.

Note: First, you must enable switch port mode private VLAN on a specified interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/pvlan/interface/<*if_type*>/<*chassis_number*>/<*if_number*>/<*is_subnumber*>
Request Body (JSON)	`{ "type" : "{type}", "op" : "{op}" "primaryVlanID" : "{pvid}" "secondaryVlanID": "{svid}" }`

where:

Element	Description
if_type	(Mandatory) The interface type (string); one of `ethernet`, for Ethernet interfaces, `po` for port-channel interfaces.
chassis_number	(Mandatory) The chassis number for the Ethernet interfaces (integer). For example: 1 for Ethernet1/2, or none for port-channel interfaces.
if_number	(Mandatory) The interface number for the Ethernet interfaces (integer). An integer from 1 to the maximum number of switch interfaces.
if_sub_number	(Optional) The sub-interface number for the Ethernet interfaces a positive integer from 1-4.
`type`	(Mandatory) PVLAN action type.
`op`	(Mandatory) Operation type; one of *add, remove*.
`primaryVlanID`	(Mandatory) The primary VLAN ID; a positive integer from *2-4093*.
`secondaryVlanID`	(Optional) The secondary VLAN ID; a positive integer from *2-4093*.

Response

Response Body (JSON)	

Show Private VLAN Information

Displays private VLAN information.

Request

Method Type	GET
Request URI	/nos/api/cfg/pvlan
Request Body (JSON)	

Response

Response Body (JSON)	<pre>{
	  “type”: “{type}”,
	  “vlanID”:”{vlanID}”,
	  “primaryVlanID”: “{primaryVlanID}”
	}</pre>

where:

Element	Description
`type`	Private VLAN type; one of *primary, isolated, community.*
`vlanID`	The VLAN ID; a positive integer from *2-4093*.
`primaryVlanID`	The primary VLAN ID; a positive integer from *2-4093*.

Show Interface Private VLAN Information

Displays the private VLAN interface information.

Request

Method Type	GET
Request URI	/nos/api/cfg/pvlan/interface
Request Body (JSON)	

Response

Response Body (JSON)	`{`
	`  “ifname”: “{ifname}”,`
	`  “vlan”:[“{vlanID}”],`
	`  “portMode”: “{portMode}”`
	`  “pvlanPortMode”:”{pvlanPortMode}”`
	`}`

where:

Element	Description
`ifname`	Interface name; one of *primary, isolated, community.*
`vlan`	The associated private VLAN; a positive integer from *2-4093*.
`portMode`	The port mode; one of *access,trunk*.
`pvlanPortMode`	The private VLAN port mode; one of *host,promiscuous,configured*.

Enable Private VLAN Globally

Globally enables the private VLAN.

Request

Method Type	POST
Request URI	/nos/api/cfg/pvlan
Request Body (JSON)	

Response

Response Body (JSON)	

Disable Private VLAN Globally

Globally disables the private VLAN.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/pvlan
Request Body (JSON)	

Response

Response Body (JSON)	

QoS over L3

The following QoS over L3 URI is available:

- /nos/api/cfg/dot1qencaps POST

The following QoS over L3 command is available:

- QoS over L3 Tag Configuration

QoS over L3 Tag Configuration

Enables or disables DOT1Q encapsulation tag value on routed port.

Request

Method Type	POST
Request URI	/nos/api/cfg/dot1qencaps

Request Body
 	
(JSON)
	{
	 “interface”: < interface_name >,
	 “dot1q_tag”: < tag_value >
	 }

where:

Element	Description
`interface`	The interface name.
`dot1q_tag`	DOT1Q tag value, an integer from *0-4093*. *0* disables the tag value.

Response

`True` if the operation succeeded; otherwise `False`.

RADIUS

The following Remote Authentication Dial-In User Service (RADIUS) URIs are available:

- /nos/api/cfg/radius GET, PUT
- /nos/api/cfg/radius/hosts GET, POST, DELETE
- /nos/api/cfg/radius/groups GET, POST, DELETE

The following RADIUS commands are available:

- Get RADIUS Configuration
- Update RADIUS Configuration
- Get RADIUS Server Configuration
- Add RADIUS Server
- Delete RADIUS Server
- Get RADIUS Server Group Configuration
- Add RADIUS Server Group
- Delete RADIUS Server Group

Get RADIUS Configuration

Gets the RADIUS configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/radius

Request Body (JSON)	{
	"source_interface":"{*source interface name*|not configured}",
	"retransmit":{*restransmit*},
	"timeout":{*timeout*},
	"global_key":"{configured|not configured}"
	}

where:

Element	Description
`source_interface`	The name of the global RADIUS source interface; one of *interface name, not configured.*
`retransmit`	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*
`timeout`	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
`global_key`	The status of the RADIUS global authentication key; one of *configured, not configured.*

Response

Response Body (JSON)	{
	"source_interface":"{*source interface name*|not configured}",
	"retransmit":{*restransmit*},
	"timeout":{*timeout*},
	"global_key":"{configured|not configured}"
	}

where:

Element	Description
`source_interface`	The name of the global RADIUS source interface; one of *interface name, not configured.*
`retransmit`	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*

Element	Description
`timeout`	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
`global_key`	The status of the RADIUS global authentication key; one of *configured, not configured.*

Update RADIUS Configuration

Updates the RADIUS configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/radius

Request Body (JSON)	{
	"retransmit":{*restransmit*},
	"timeout":{*timeout*},
	"global_key":"{*global_key*}",
	"global_key_form":{0|7},
	"source_interface":"{*source interface name*|default}"
	}

where:

Element	Description
`retransmit`	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*
`timeout`	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
`global_key`	The status of the RADIUS global authentication key; one of *configured, not configured.*
`global_key_form`	The encryption method of the RADIUS global authentication key; one of *0* (clear text), *7* (encrypted).
`source_interface`	The name of the global RADIUS source interface; one of *interface name, default.* **Note:** If the variable is set to *default*, then it will reset to *not configured*.

Response

Response Body (JSON)	{
	"source_interface":"{*source interface name*|not configured}",
	"retransmit":{*restransmit*},
	"timeout":{*timeout*},
	"global_key":"{configured|not configured}"
	}

where:

Element	Description
`source_interface`	The name of the global RADIUS source interface; one of *interface name, not configured.*
`retransmit`	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*
`timeout`	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
`global_key`	The status of the RADIUS global authentication key; one of *configured, not configured.*

Get RADIUS Server Configuration

Gets the configuration of a specific configured RADIUS server or of all configured RADIUS servers.

Request

Method Type	GET
Request URI	/nos/api/cfg/radius/hosts/<*IP_addr*>
Request Body (JSON)	

where:

Element	Description
IP_addr	The IP address of the RADIUS server. **Note:** If no IP address is provided, then the command returns the configuration of all configured RADIUS servers.

Response

Response Body (JSON)	`[`
	`  {`
	`  "IP_addr":"{`*IP_addr*`}",`
	`  "retransmit":{`*retransmit*`},`
	`  "timeout":{`*timeout*`},`
	`  "key":"{configured|not configured}",`
	`  "auth-port":{`*auth-port*`},`
	`  "acct-port":{`*acct-port*`}`
	`  }`
	`]`

where:

Element	Description
`IP_addr`	The IP address of the RADIUS server.
`retransmit`	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*
`timeout`	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
`key`	The status of the RADIUS server authentication key; one of *configured, not configured.*

Element	Description
`auth-port`	The TCP port used for RADIUS server authentication; an integer from *0-65535.*
`acct-port`	The TCP port used for RADIUS server accounting; an integer from *0-65535.*

Add RADIUS Server

Configures a RADIUS server.

Request

Method Type	POST
Request URI	/nos/api/cfg/radius/hosts

Request Body (JSON)	{
	"IP_addr":"{*IP_addr*}",
	"retransmit":{*retransmit*},
	"timeout":{*timeout*},
	"key":"{*key*}",
	"key_form":{0|7},
	"auth-port":{*auth-port*},
	"acct-port":{*acct-port*}
	}

where:

Element	Description
IP_addr	The IP address of the RADIUS server.
retransmit	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*
timeout	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
key	The RADIUS server authentication key (string).
key_form	The encryption method of the RADIUS server authentication key; one of *0* (clear text), *7* (encrypted).
auth-port	The TCP port used for RADIUS server authentication; an integer from *0-65535.*
acct-port	The TCP port used for RADIUS server accounting; an integer from *0-65535.*

Response

Response Body (JSON)	{
	"IP_addr":"{*IP_addr*}",
	"retransmit":{*retransmit*},
	"timeout":{*timeout*},
	"key":"{configured|not configured}",
	"auth-port":{*auth-port*},
	"acct-port":{*acct-port*}
	}

where:

Element	Description
IP_addr	The IP address of the RADIUS server.
retransmit	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*
timeout	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
key	The status of the RADIUS server authentication key; one of *configured, not configured.*
auth-port	The TCP port used for RADIUS server authentication; an integer from *0-65535.*
acct-port	The TCP port used for RADIUS server accounting; an integer from *0-65535.*

Delete RADIUS Server

Removes a configured RADIUS server.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/radius/hosts/<*IP_addr*>
Request Body (JSON)	

where:

Element	Description
IP_addr	The IP address of the RADIUS server.

Get RADIUS Server Group Configuration

Gets the configuration of a specific configured RADIUS server group or of all configured RADIUS server groups.

Request

Method Type	GET
Request URI	/nos/api/cfg/radius/groups/<*group_name*>
Request Body (JSON)	

where:

Element	Description
group_name	The name of the RADIUS server group; a string up to 127 characters long. **Note:** If no group name is provided, then the command returns the configuration of all configured RADIUS server groups.

Response

Response Body (JSON)

```
[
   {
      "group_name":"{group_name}"
      "vrf_name":"{vrf_name}",
      "source_interface":"{source interface name|not configured}",
      "hosts":
         [
            {
               "IP_addr":"{IP_addr}",
               "retransmit":{retransmit},
               "timeout":{timeout},
               "key":"{configured|not configured}",
               "auth-port":{auth-port},
               "acct-port":{acct-port}
            }
         ]
   }
]
```

where:

Element	Description
`group_name`	The name of the RADIUS server group; a string up to 127 characters long.
`vrf_name`	The VRF instance for the RADIUS server group. Valid value: the *VRF instance name.*
`source_interface`	The name of the RADIUS group source interface; one of *interface name, not configured.*
`hosts`	The list of servers members of the RADIUS server group.
`IP_addr`	The IP address of the RADIUS server.
`retransmit`	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*
`timeout`	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
`key`	The status of the RADIUS server authentication key; one of *configured, not configured.*
`auth-port`	The TCP port used for RADIUS server authentication; an integer from *0-65535.*
`acct-port`	The TCP port used for RADIUS server accounting; an integer from *0-65535.*

Add RADIUS Server Group

Configures a RADIUS server group.

Request

Method Type	POST
Request URI	/nos/api/cfg/radius/groups

Request Body (JSON)	{
	"group_name":"{group_name}",
	"vrf_name":"{vrf_name}",
	"source_interface":"{*source interface name*|default}",
	"hosts":
	[
	"{*IP_addr*}"
	]
	}

where:

Element	Description
`group_name`	The name of the RADIUS server group; a string up to 127 characters lonh.
`vrf_name`	The VRF instance for the RADIUS server group. Valid value: the *VRF instance name.*
`source_interface`	The name of the RADIUS group source interface; one of *interface name, default.* **Note:** If the variable is set to *default*, then it will reset to *not configured*.
`hosts`	The list of servers members of the RADIUS server group.
IP_addr	The IP address of the RADIUS server to be added to the group.

Response

Response Body (JSON)	[
	{
	"group_name":"{*group_name*}"
	"vrf_name":"{*vrf_name*}",
	"source_interface":"{*source interface name*|not configured}",
	"hosts":
	[
	{
	"IP_addr":"{*IP_addr*}",
	"retransmit":{*retransmit*},
	"timeout":{*timeout*},
	"key":"{configured|not configured}",
	"auth-port":{*auth-port*},
	"acct-port":{*acct-port*}
	}
	]
	}
	]

where:

Element	Description
`group_name`	The name of the RADIUS server group; a string up to 127 characters long.
`vrf_name`	The VRF instance for the RADIUS server group. Valid value: the *VRF instance name.*
`source_interface`	The name of the RADIUS group source interface; one of *interface name, not configured.*
`hosts`	The list of servers members of the RADIUS server group.
`IP_addr`	The IP address of the RADIUS server.
`retransmit`	The number of retries the switch will make to establish a connection with a RADIUS server after the initial attempt failed; an integer from *0-5.*
`timeout`	The amount of time, in seconds, before a RADIUS server connection attempt is considered to have failed; an integer from *1-60.*
`key`	The status of the RADIUS server authentication key; one of *configured, not configured.*
`auth-port`	The TCP port used for RADIUS server authentication; an integer from *0-65535.*
`acct-port`	The TCP port used for RADIUS server accounting; an integer from *0-65535.*

Delete RADIUS Server Group

Removes a configured RADIUS server group.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/radius/groups/<*group_name*>
Request Body (JSON)	

where:

Element	Description
group_name	The name of the RADIUS server group; a string up to 127characters long.

REST

The following REST URIs are available:

- /nos/api/cfg/rest/server GET, PUT

The following REST commands are available:

- Get REST Server
- Set REST Server

Get REST Server

Gets the REST server status.

Request

Method Type	GET
Request URI	/nos/api/cfg/rest/server
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	    "status": "{enable | disable}",
	    "protocol": "{http | https}",
	}</pre>

where:

Element	Description
`status`	The REST server status (string); one of *enable, disable*.
`protocol`	The REST server protocol; one of *http, https*.

Set REST Server

Sets the REST server status.

Request

Method Type	PUT
Request URI	/nos/api/cfg/rest/server

Request Body
 	
(JSON)
	<pre>{
	    "status": "{enable | disable}",
	    "protocol": "{enable | disable}",
	}</pre>

where:

Element	Description
`status`	The REST server status (string); one of *enable, disable*.
`protocol`	The REST server protocol; one of *http, https*.

Response

Response Body (JSON)	

Route Maps

The following route map URI is available:

- /nos/api/info/routemap GET

The following route map command is available:

- Get Route Maps

Get Route Maps

Gets route maps property information.

Request

Method Type	GET
Request URI	/nos/api/info/routemap
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	{
	"rmapName": "*<rmapName>*",
	"matchAction": "*<matchAction>*",
	"seqNum": "*<seqNum>*",
	"matchList":
	[
	{
	“cmd”: "*<matchList_cmd>*"
	}
	],
	"applyList":
	[
	{
	“cmd”: "*<cmd>*"
	}
	]
	}
	]

where:

Element	Description
`rmapName`	Route map name
`matchAction`	Match action
`seqnum`	Sequence number
`matchList_cmd`	Match list command
`applyList_cmd`	Apply list command

Security Mode

The following security mode URI is available:

- /nos/api/cfg/secmode GET, PUT

The following security mode commands are available:

- Get Security Mode Configuration
- Update Security Mode Configuration

Get Security Mode Configuration

Gets the current security mode configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/secmode
Request Body (JSON)	

Response

Response Body (JSON)	`{`
	`  "current":"{legacy_mode|secure_mode}",`
	`  "setting":"{legacy_mode|secure_mode}"`
	`}`

where:

Element	Description
`current`	The currently configured security mode. Valid values: • *legacy_mode* • *secure_mode*
`setting`	The security mode that takes effect after switch reload. Valid values: • *legacy_mode* • *secure_mode*

Update Security Mode Configuration

Updates the security mode configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/secmode

Request Body (JSON)	`{`
	`  "setting":"{legacy_mode|secure_mode}"`
	`}`

where:

Element	Description
`setting`	The security mode that takes effect after switch reload; one of *legacy_mode, secure_mode.*

Response

Response Body (JSON)	`{`
	`  "current":"{legacy_mode|secure_mode}",`
	`  "setting":"{legacy_mode|secure_mode}"`
	`}`

where:

Element	Description
`current`	The currently configured security mode; one of *legacy_mode, secure_mode.*
`setting`	The security mode that takes effect after switch reload; one of *legacy_mode, secure_mode.*

sFlow

The following sFlow URIs are available:

- /nos/api/cfg/sflow GET, PUT
- /nos/api/cfg/sflow/interface/<*if_name*> PUT
- /nos/api/info/sflow/stats GET, DELETE

The following sFlow commands are available:

- Get sFlow Configuration
- Update sFlow Global Configuration
- Update sFlow Interface Configuration
- Get sFlow Statistics
- Clear sFlow Statistics

Get sFlow Configuration

Gets the current sFlow configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/sflow
Request Body (JSON)	

Response

Response Body (JSON)	{
	  enabled: [enabled|disabled],
	  sampling-rate: <*sampling_rate*>,
	  max-sampled-size: <*sampled_size*>,
	  polling-interval: <*polling_interval*>,
	  max-datagram-size: <*datagram_size*>,
	  collector:
	  {
	  ip: <*ip_addr*>,
	  port: <*port*>,
	  vrf: <*vrf*>
	  },
	  interfaces:
	  [
	  <*if_name*>,
	  ...
	  ]
	}

where:

Element	Description
`enabled`	The status of sFlow on the switch; one of *enabled, disabled.*
`sampling-rate`	The sFlow sampling rate, in packets per sample; an integer from *4096-1000000000.* Default value: *4096.*
`max-sampled-size`	The maximum sFlow sampling size of packets; an integer from *64-256.* Default value: *128.*
`polling-interval`	The sFlow polling interval, in seconds; an integer from *0-86400.* **Note:** Setting this variable to *0* disables sFlow polling.
`max-datagram-size`	The maximum sFlow datagram size; an integer from *200-9000.* Default value: *1500*
`collector`	The sFlow collector configuration.
`ip`	The IP address of the sFlow server.

Element	Description
`port`	The UDP port of the sFlow server; an integer from *1-65535.* Default value: *6343*
`vrf`	The VRF instance used by the sFlow server. Valid value: the *VRF instance name.*
`interfaces`	The interfaces on which sFlow sampling is enabled. Valid value: the *interface name* (for example, *Ethernet1/12*).

Update sFlow Global Configuration

Updates the global sFlow configuration on the switch.

Request

Method Type	PUT
Request URI	/nos/api/cfg/sflow

Request Body (JSON)	{
	  enabled: [enabled|disabled],
	  sampling-rate: <*sampling_rate*>,
	  max-sampled-size: <*sampled_size*>,
	  polling-interval: <*polling_interval*>,
	  max-datagram-size: <*datagram_size*>,
	  collector:
	  {
	  ip: <*ip_addr*>,
	  port: <*port*>,
	  vrf: <*vrf*>
	  },
	}

where:

Element	Description
`enabled`	The status of sFlow on the switch; one of *enabled, disabled.*
`sampling-rate`	The sFlow sampling rate, in packets per sample; an integer from *4096-1000000000.* Default value: *4096.*
`max-sampled-size`	The maximum sFlow sampling size of packets; an integer from *64-256.* Default value: *128.*
`polling-interval`	The sFlow polling interval, in seconds; an integer from *0-86400.* **Note:** Setting this variable to *0* disables sFlow polling.
`max-datagram-size`	The maximum sFlow datagram size; an integer from *200-9000.* Default value: *1500*
`collector`	The sFlow collector configuration.
`ip`	The IP address of the sFlow server.
`port`	The UDP port of the sFlow server. Valid value: • *1-65535* Default value: *6343*
`vrf`	The VRF instance used by the sFlow server. Valid value: • *VRF instance name*

Update sFlow Interface Configuration

Updates the sFlow configuration for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/sflow/interface/<*if_name*>

Request Body (JSON)	{
	  enabled: [enabled|disabled]
	}

where:

Element	Description
if_name	The name of the switch interface. For example, *Ethernet1/12*.
`enabled`	The status of sFlow on the switch; one of *enabled, disabled.*

Get sFlow Statistics

Gets sFlow statistics.

Request

Method Type	GET
Request URI	/nos/api/info/sflow/stats
Request Body (JSON)	

Response

Response Body (JSON)	{
	`processed_samples:` *<processed_samples>*,
	`dropped_samples:` *<dropped_samples>*,
	`sent_datagrams:` *<sent_datagrams>*,
	`dropped_datagrams:` *<dropped_datagrams>*
	}

where:

Element	Description
`processed_samples`	The number of processed sFlow samples (integer). Valid value: • *integer*
`dropped_samples`	The number of dropped sFlow samples. (integer).
`sent_datagrams`	The number of sent sFlow datagrams (integer).
`dropped_datagrams`	The number of dropped sFlow datagrams (integer).

Clear sFlow Statistics

Resets sFlow statistics.

Request

Method Type	DELETE
Request URI	/nos/api/info/sflow/stats/clear
Request Body (JSON)	

SNMP

The following SNMP-related URIs are available:

- /nos/api/cfg/snmp/hosuser GET, PUT, DELETE
- /nos/api/hostraphost GET, PUT, DELETE

Note: These URIs and commands are necessary for XClarity support.

The following SNMP commands are available:

- Get the SNMPv3 Account for XClarity
- Set the SNMPv3 Account for XClarity
- Delete the SNMPv3 Account for XClarity
- Get the SNMPv3 Trap Host IP Address for XClarity
- Set the SNMPv3 Trap Host IP Address for XClarity
- Delete the SNMPv3 Trap Host IP Address for XClarity

Get the SNMPv3 Account for XClarity

Gets the special SNMPv3 user account for XClarity.

Request

Method Type	GET
Request URI	/nos/api/cfg/snmp/hosuser
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"user_name": "*<user_name>*",
	"group_name":"*<group_name>*",
	"auth_type":"*<auth_type>*",
	"auth_passwd":"*<password>*",
	"priv_type":" "*<priv_type>*",
	"priv_passwd":"*<password>*"
	"xclarity_id": "*<identifier>*"
	}

where:

Element	Description
`user_name`	User name; a text string from 5-32 characters long.
`group_name`	Group name; one of *network_operator, network_admin*. Default value: *network-operator*.
`auth_type`	Authentication type; one of *invalid, md5, sha*.
`auth_passwd`	Authentication password; a string from 8-32 characters long.
`priv_type`	Privilege type; one of *invalid, des, aes*.
`priv_passwd`	Privilege password; a string from 8-32 characters long.
`xclarity_id`	XClarity instance identifier (string).

Note: If the user account does not exist, the request will return the error message "`404 Resource not available: Snmp Hos User instance not present in DUT`".

Set the SNMPv3 Account for XClarity

Sets the special SNMPv3 user account for XClarity.

Note: This account is only for XClarity support and differs from SNMPv3 user accounts in the following ways:

- It cannot be set from the command line interface.
- It does not appear in any command.
- This information must be included in any tech support dump.

Request

Method Type	PUT
Request URI	/nos/api/cfg/snmp/hosuser

Request Body
 	
(JSON)
	{
	"user_name": "*<user_name>*",
	"group_name":"*<group_name>*",
	"auth_type":"*<auth_type>*",
	"auth_passwd":"*<password>*",
	"priv_type":" "*<priv_type>*",
	"priv_passwd":"*<password>*"
	"xclarity_id": "*<identifier>*"
	}

where:

Element	Description
`user_name`	User name; a text string from 5-32 characters long.
`group_name`	Group name; one of *network_operator, network_admin*. Default value: *network-operator*.
`auth_type`	Authentication type; one of *invalid, md5, sha*.
`auth_passwd`	Authentication password; a string from 8-32 characters long.
`priv_type`	Privilege type; one of *invalid, des, aes*.
`priv_passwd`	Privilege password; a string from 8-32 characters long.
`xclarity_id`	XClarity instance identifier (string).

Response

Response Body
 	
(JSON)
	{
	"username": "*<username>*",
	"group_name":"*<group_name>*",
	"auth_type":"*<auth_type>*",
	"auth_passwd":"*<password>*",
	"priv_type":" "*<priv_type>*",
	"priv_passwd":"*<password>*"
	"xclarity_id": "*<identifier>*"
	}

where:

Element	Description
`user_name`	User name; a text string from 5-32 characters long.
`group_name`	Group name; one of *network_operator, network_admin*. Default value: *network-operator*.
`auth_type`	Authentication type; one of *invalid, md5, sha*.
`auth_passwd`	Authentication password; a string from 8-32 characters long.
`priv_type`	Privilege type; one of *invalid, des, aes*.
`priv_passwd`	Privilege password; a string from 8-32 characters long.
`xclarity_id`	XClarity instance identifier (string).

Delete the SNMPv3 Account for XClarity

Deletes the special SNMPv3 user account for XClarity.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/snmp/hosuser
Request Body (JSON)	

Response

`True` if the operation succeeded; otherwise `False`.

Get the SNMPv3 Trap Host IP Address for XClarity

Gets the IP address of the special SNMPv3 trap host for XClarity.

Request

Method Type	GET
Request URI	/nos/api/cfg/snmp/hostraphost
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"host_name": "*<host_name>*",
	"security_level":"*<security_level>*",
	"username":"*<username>*",
	"message_type":" "*<priv_type>*",
	"port":"*<port>*"
	}

where:

Element	Description
`host_name`	Trap host name; a valid IPv4 or IPv6 address.
`security_level`	Security level; one of *auth, authpriv, noauth*.
`username`	Username; a text string 5-32 characters long.
`message_type`	Message type; one of *trap, inform*. Default value: *trap*.
`port`	Host UDP port; an integer from *1-65535*. Default value: *162*.

Note: If the user name does not exist, the request will return the error message "`404 Resource not available: Snmp Hos User instance not present in DUT`".

Set the SNMPv3 Trap Host IP Address for XClarity

Sets the IP address of the special SNMPv3 trap host for XClarity.

Note: The trap destination IP address is only for XClarity and differs from a standard SNMPv3 trap host in the following ways:

- It cannot be set from the command line interface.
- It does not appear in any command.
- This information must be included in any tech support dump.

Request

Method Type	PUT
Request URI	/nos/api/cfg/snmp/hostraphost

Request Body
 	
(JSON)
	{
	"host_name": "*<host_name>*",
	"security_level":"*<security_level>*",
	"username":"*<username>*",
	"message_type":" "*<priv_type>*",
	"port":"*<port>*"
	}

where:

Element	Description
`host_name`	Trap host name; a valid IPv4 or IPv6 address.
`security_level`	Security level; one of *auth, authpriv, noauth*.
`username`	Username; a text string 5-32 characters long.
`message_type`	Message type; one of *trap, inform*. Default value: *trap*.
`port`	Host UDP port; an integer from *1-65535*. Default value: *162*.

Response

Response Body
 	
(JSON)
	{
	"host_name": "*<host_name>*",
	"security_level":"*<security_level>*",
	"username":"*<username>*",
	"message_type":" "*<priv_type>*",
	"port":"*<port>*"
	}

where:

Element	Description
`host_name`	Trap host name; a valid IPv4 or IPv6 address.
`security_level`	Security level; one of *auth, authpriv, noauth*.
`username`	Username; a text string 5-32 characters long.
`message_type`	Message type; one of *trap, inform*. Default value: *trap*.
`port`	Host UDP port; an integer from *1-65535*. Default value: *162*.

Delete the SNMPv3 Trap Host IP Address for XClarity

Deletes the special SNMPv3 trap host IP address for XClarity.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/snmp/hostraphost
Request Body (JSON)	

Response

`True` if the operation succeeded; otherwise `False`.

SSH

The following Secure Shell (SSH) URIs are available:

- /nos/api/cfg/ssh/server GET, PUT

The following SSH commands are available:

- Get SSH Server
- Set SSH Server

Get SSH Server

Gets the SSH server status.

Request

Method Type	GET
Request URI	/nos/api/cfg/ssh/server
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"status": "{enable | disable}"
	}

where:

Element	Description
`status`	The SSH server status (string); one of *enable, disable.*

Set SSH Server

Sets the SSH server status.

Request

Method Type	PUT
Request URI	/nos/api/cfg/ssh/server

Request Body
(JSON)
	`{`
	`  "status": "{enable/disable}"`
	`}`

where:

Element	Description
`status`	The SSH server status (string); one of *enable, disable.*

Response

Response Body
(JSON)
	`{`
	`  "status": "{enable | disable}"`
	`}`

where:

Element	Description
`status`	The SSH server status (string); one of *enable, disable.*

Startup Information

The following startup information URIs are available:

- /nos/api/startup GET
- /nos/api/startup/ztp GET, PUT
- /nos/api/startup/software GET, PUT

The following startup commands are available:

- Get System Startup Information
- Put System ZTP Interface
- Get System ZTP Setting
- Put System Startup Image
- Get System Startup Image

Get System Startup Information

Gets system boot information.

Note: This is required for XClarity support.

Request

Method Type	GET
Request URI	/nos/api/startup
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	  “ztp” : “Forcedly Enabled”,
	  “active image” : “version 0.0.0.0, downloaded 19:20:29 UTC Tue Feb 16 2016”,
	  “standby image” : “version 0.0.0.0, downloaded 13:20:02 UTC Sun Feb 28 2016”,
	  “Uboot” : “version 0.0.0.0, downloaded 13:20:04 UTC Sun Feb 28 2016”,
	  “ONIE” : “empty”,
	  “boot software” : “active”,
	  “scheduled reboot”: “none”,
	  “port mode” : “default”
	}</pre>

where:

Element	Description
`ztp`	Current zero touch provisioning setting.
`active image`	Active image information.
`standby image`	Standby image information.
`Uboot`	Uboot image information.
`ONIE`	ONIE image information.
`boot software`	Next boot image setting.
`scheduled reboot`	Scheduled reboot setting.
`port mode`	Current port mode.

Put System ZTP Interface

Updates the system Zero Touch Provisioning setting.

Request

Method Type	PUT
Request URI	/nos/api/startup/ztp/<*enable*>

Request Body
 	
(JSON)
	{
	  "ztp" : "<*setting*>"
	}

where:

Element	Description
ztp	The ZTP provisioning setting; one of *Enable, Forcedly Enabled , Forcedly Disabled*.

Response

Response Body
 	
(JSON)
	{
	  "ztp" : "<*setting*>"
	}

where:

Element	Description
ztp	Current ZTP provisioning setting.

Get System ZTP Setting

Gets the current system Zero Touch Provisioning setting.

Request

Method Type	GET
Request URI	/nos/api/startup/ztp
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  "ztp" : "*<setting>*"
	}

where:

Element	Description
`ztp`	Current ZTP provisioning setting.

Put System Startup Image

Updates the system startup image.

Note: This request is required for XClarity support.

Request

Method Type	PUT
Request URI	/nos/api/startup/software

Request Body
 	
(JSON)
	{
	  "boot software" : "*<setting>*"
	}

where:

Element	Description
`boot software`	Next startup image setting; one of *active, standby.*

Response

Response Body
 	
(JSON)
	{
	  "boot software" : "*<setting>*"
	}

where:

Element	Description
`boot software`	Next startup image setting; one of *active, standby.*

Get System Startup Image

Gets the system boot image.

Note: This request is required for XClarity support.

Request

Method Type	GET
Request URI	/nos/api/startup/software
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  “boot software” : “*<setting>*”
	}

where:

Element	Description
`boot software`	Next startup image setting; one of *active*, *standby*.

STP

The following STP URIs are available:

- /nos/api/cfg/stp_interface GET
- /nos/api/cfg/stp_interface/<*if_name*> GET, PUT
- /nos/api/cfg/stp/vlan/<*vid*> GET, PUT
- /nos/api/cfg/stp/interface/<*if_name*> GET, PUT
- /nos/api/cfg/stp/interface/<*if_name*>/vlan/<*vlan_id*> GET, PUT

The following STP interface property commands are available:

- Get STP Properties for All Interfaces
- Get STP Interface Properties
- Update STP Interface Properties
- Get STP Properties Per VLAN
- Set STP Properties Per VLAN
- Get STP Interface Properties
- Update STP Interface Properties
- Get STP Interface VLAN Properties
- Update STP Interface VLAN Properties

Get STP Properties for All Interfaces

Gets STP properties of all interfaces. These properties are supported by all STP modes.

Request

Method Type	GET
Request URI	/nos/api/cfg/stp/interface
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"if_name": "*<if_name>*",
	"edge_port": "*<edge_port>*",
	"bpdu_guard": "*<bpdu_guard>*",
	"loop_guard": "*<loop_guard>*",
	"root_guard": "*<root_guard>*"
	}

where:

Element	Description
`if_name`	The IP interface name (string). **Note:** The interface must exist.
`edge_port`	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of *yes, no.* Default value: *yes*.
`bpdu_guard`	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of *enable, disable*. Default value: *disable*.
`loop_guard`	(Optional) Whether look guard is enabled on a port for additional checks for preventing STP looping; one of *enable, disable*. Default value: *disable*.
`root_guard`	(Optional) Whether guard mode is set to root guard on interface.

Get STP Interface Properties

Gets STP properties of one interface. These properties are supported by all STP modes.

Request

Method Type	GET
Request URI	/nos/api/cfg/stp/interface/<*if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"edge_port": "<*edge_port*>",
	"bpdu_guard": "<*bpdu_guard*>",
	"loop_guard": "<*loop_guard*>",
	"root_guard": "<*root_guard*>"
	}

where:

Element	Description
if_name	The IP interface name (string). **Note:** The interface must exist.
`edge_port`	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of *yes, no.* Default value: *yes*.
`bpdu_guard`	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of *enable, disable*. Default value: *disable*.
`loop_guard`	(Optional) Whether look guard is enabled on a port for additional checks for preventing STP looping; one of *enable, disable*. Default value: *disable*.
`root_guard`	(Optional) Whether guard mode is set to root guard on interface.

Update STP Interface Properties

Updates STP properties of one interface. These properties are supported by all STP modes.

Note: If an element is not specified in a PUT request, no update for that element will be performed.

Request

Method Type	PUT
Request URI	/nos/api/cfg/stp/interface/<*if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"edge_port": "<*edge_port*>",
	"bpdu_guard": "<*bpdu_guard*>",
	"loop_guard": "<*loop_guard*>",
	"root_guard": "<*root_guard*>"
	}

where:

Element	Description
if_name	The IP interface name (string). **Note:** The interface must exist.
`edge_port`	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of *yes, no*. Default value: *yes*.
`bpdu_guard`	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of *enable, disable*. Default value: *disable*.
`loop_guard`	(Optional) Whether look guard is enabled on a port for additional checks for preventing STP looping; one of *enable, disable*. Default value: *disable*.
`root_guard`	(Optional) Whether guard mode is set to root guard on interface.

Response

Response Body
 	
(JSON)
	{
	"if_name": "*<if_name>*",
	"edge_port": "*<edge_port>*",
	"bpdu_guard": "*<bpdu_guard>*",
	"loop_guard": "*<loop_guard>*",
	"root_guard": "*<root_guard>*"
	}

where:

Element	Description
`if_name`	The IP interface name (string). **Note:** The interface must exist.
`edge_port`	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of *yes, no*. Default value: *yes*.
`bpdu_guard`	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of *enable, disable*. Default value: *disable*.
`loop_guard`	(Optional) Whether look guard is enabled on a port for additional checks for preventing STP looping; one of *enable, disable*. Default value: *disable*.
`root_guard`	(Optional) Whether guard mode is set to root guard on interface.

Get STP Properties Per VLAN

Gets STP parameters for each VLAN.

Note: This is designed only for rapid Per VLAN Spanning Tree (PVST) mode.

Request

Method Type	GET
Request URI	/nos/api/cfg/stp/vlan/<*vlan_ID*>
Request Body (JSON)	

where:

Element	Description
vlan_ID	VLAN ID; an integer from 2-3999.

Response

Response Body
 	
(JSON)
	{
	"forward-time": "<*forward-time*>",
	"hello-time": "<*hello-time*>",
	"max-age" : "<*max-age*>",
	"priority" : "<*priority*>"
	}

where:

Element	Description
`forward-time`	The forward delay for the spanning tree; an integer from *4-30*.
`hello-time`	The hello interval for the spanning tree; an integer from *1-10*.
`max-age`	The maximum age interval for the spanning tree; an integer from *6-40*.
`priority`	The bridge priority for the spanning tree; an integer from *0-61440*.

Set STP Properties Per VLAN

Sets STP parameters for each VLAN.

Note: This is designed only for rapid Per VLAN Spanning Tree (PVST) mode.

Request

Method Type	PUT
Request URI	/nos/api/cfg/stp/vlan/<*vlan_ID*>

Request Body
 	
(JSON)
	{
	"forward-time": "<*forward-time*>",
	"hello-time": "<*hello-time*>",
	"max-age" : "<*max-age*>",
	"priority" : "<*priority*>"
	}

where:

Element	Description
vlan_ID	VLAN number.; an integer from 2-3999.
`forward-time`	The forward delay for the spanning tree; an integer from *4-30*.
`hello-time`	The hello interval for the spanning tree; an integer from *1-10*.
`max-age`	The maximum age interval for the spanning tree; an integer from *6-40*.
`priority`	The bridge priority for the spanning tree; an integer from *0-61440*.

Response

Response Body
 	
(JSON)
	{
	"forward-time": "<*forward-time*>",
	"hello-time": "<*hello-time*>",
	"max-age" : "<*max-age*>",
	"priority" : "<*priority*>"
	}

where:

Element	Description
`forward-time`	(Optional) The forward delay for the spanning tree; an integer from *4-30*.
`hello-time`	(Optional) The hello interval for the spanning tree; an integer from *1-10*.

Element	Description
`max-age`	(Optional) The maximum age interval for the spanning tree; an integer from *6-40*.
`priority`	(Optional) The bridge priority for the spanning tree; an integer from *0-61440*.

Get STP Interface Properties

Gets the STP properties of the specified interface.

Note: These properties are supported by all STP modes.

Request

Method Type	GET
Request URI	/nos/api/cfg/stp/interface/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The IP interface name (string). **Note:** The interface must exist.

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"edge_port": "<*edge_port*>",
	"bpdu_guard": "<*bpdu_guard*>",
	"loop_guard": "<*loop_guard*>",
	"root_guard": "<*root_guard*>"
	}

where:

Element	Description
`if_name`	The IP interface name (string). **Note:** The interface must exist.
`edge_port`	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of *yes, no.* Default value: *yes*.
`bpdu_guard`	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of *enable, disable*. Default value: *disable*.
`loop_guard`	(Optional) Whether look guard is enabled on a port for additional checks for preventing STP looping; one of *enable, disable*. Default value: *disable*.
`root_guard`	(Optional) Whether guard mode is set to root guard on interface.

Update STP Interface Properties

Updates the STP properties of the specified interface.

Note: These properties are supported by all STP modes.

Request

Method Type	PUT
Request URI	/nos/api/cfg/stp/interface/<*if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"edge_port": "<*edge_port*>",
	"bpdu_guard": "<*bpdu_guard*>",
	"loop_guard": "<*loop_guard*>",
	"root_guard": "<*root_guard*>"
	}

where:

Element	Description
if_name	The IP interface name (string). **Note:** The interface must exist.
`edge_port`	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of *yes, no*. Default value: *yes*.
`bpdu_guard`	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of *enable, disable*. Default value: *disable*.
`loop_guard`	(Optional) Whether look guard is enabled on a port for additional checks for preventing STP looping; one of *enable, disable*. Default value: *disable*.
`root_guard`	(Optional) Whether guard mode is set to root guard on interface.

Get STP Interface VLAN Properties

Gets the STP interface VLAN properties of the specified interface and VLAN.

Note: This is designed only for rapid Per VLAN Spanning Tree (PVST) mode.

Request

Method Type	GET
Request URI	/nos/api/cfg/stp/interface/<*if_name*>/vlan/<*vlan_id*>
Request Body (JSON)	

where:

Element	Description
if_name	The IP interface name (string). **Note:** The interface must exist.
vlan_ID	VLAN ID; an integer from 2-3999.

Response

Response Body
 	
(JSON)
	{
	"cost": "<*cost*>",
	"priority": "<*priority*>",
	}

where:

Element	Description
`cost`	The interface's spanning-tree port path cost; one of *auto* (based on port speed), an integer from *1-200000000*.
`priority`	The interface's spanning-tree port path priority, in increments of 32; an integer from *0-224* that is a multiple of 32. Default value: *128*.

Update STP Interface VLAN Properties

Updates the STP interface VLAN properties of the specified interface and VLAN.

Note: This is designed only for rapid Per VLAN Spanning Tree (PVST) mode.

Request

Method Type	PUT
Request URI	/nos/api/cfg/stp/interface/<*if_name*>/vlan/<*vlan_id*>

Request Body
 	
(JSON)
	{
	"cost": "<*cost*>",
	"priority": "<*priority*>",
	}

where:

Element	Description
if_name	The IP interface name (string). **Note:** The interface must exist.
vlan_ID	VLAN ID; an integer from 2-3999.
`cost`	The interface's spanning-tree port path cost; one of *auto* (based on port speed), an integer from *1-200000000*.
`priority`	The interface's spanning-tree port path priority, in increments of 32; an integer from *0-224* that is a multiple of 32. Default value: *128*.

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"edge_port": "<*edge_port*>",
	"bpdu_guard": "<*bpdu_guard*>",
	}

where:

Element	Description
`if_name`	The IP interface name (string). **Note:** The interface must exist.
`edge_port`	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of *yes, no*. Default value: *yes*.
`bpdu_guard`	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of *enable, disable*. Default value: *disable*.

System

The following system URI is available:

- /nos/api/system GET

The following system interface property command is available:

- Get System Properties

Get System Properties

Gets basic properties of the system. All properties are version-independent.

Request

Request URI	`/nos/api/system`
Request Body (JSON)	

Response

```
{
  "switch_type": "{switch_type}",
  "fw_version": "{version}"
}
```

where:

Element	Description
`switch_type`	Switch platform type.
`fw_version`	The version number of the firmware running on the switch.

System Configuration

The following system configuration-related URIs are available:

- /nos/api/cfg/hostname GET, PUT
- /nos/api/cfg/clock GET, PUT
- /nos/api/cfg/clock/format PUT
- /nos/api/cfg/clock/protocol PUT
- /nos/api/cfg/clock/timezone PUT
- /nos/api/cfg/clock/summertime PUT
- /nos/api/cfg/syscontact GET, PUT
- /nos/api/cfg/sysdescr GET, PUT
- /nos/api/cfg/rack_prop GET, PUT

Note: These requests are required for XClarity support.

The following system configuration commands are available:

- Get Hostname
- Set Hostname
- Get Clock Date
- Set Clock Date
- Set Clock Format
- Set Clock Protocol
- Set Clock Timezone
- Set Clock Summer Time
- Get Device Contact
- Update Device Contact
- Get Device Description
- Update Device Description
- Get Rack Properties
- Update Rack Properties

Get Hostname

Gets the hostname of the system.

Request

Method Type	GET
Request URI	/nos/api/cfg/hostname
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  "hostname": "*<hostname>*",
	}

where:

Element	Description
`hostname`	The hostname of the system; a string from 1-64 characters long.

Set Hostname

Sets the hostname of the system.

Request

Method Type	PUT
Request URI	/nos/api/cfg/hostname
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  "hostname": "*<hostname>*",
	}

where:

Element	Description
`hostname`	The hostname of the system; a string from 1-64 characters long.

Get Clock Date

Gets the system date.

Request

Method Type	GET
Request URI	/nos/api/cfg/clock
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  "date": "*<date>*",
	}

where:

Element	Description
`date`	System date in the following format: HH:MM:SS xM ZZZ Wkd Mon Dy YEAR where: • HH - hour • MM - minutes • SS - seconds • xM - one of "`AM`", "`PM`" • ZZZ - name of the time zone • Wkd - three-letter weekday abbreviation • Mon - three-letter month abbreviation • Dy - one or two-digit day • YEAR - four-digit year For example: `10:55:58 AM UTC Mon Jul  4 2016`

Set Clock Date

Sets the system date and time.

Request

Method Type	PUT
Request URI	/nos/api/cfg/clock

Request Body
 	
(JSON)
	{
	"time": "<*HH*:*MM*:*SS*>" ,
	"day": <*day*>,
	"month": <*month*> ,
	"year": <*year*>
	}

where:

Element	Description
`time`	System time in the following format: HH:MM:SS.
`day`	The day of the month; an integer from *1-31*.
`month`	The month; one of the following case-insensitive strings: • *January* • *February* • *March* • *April* • *May* • *June* • *July* • *August* • *September* • *October* • *November* • *December*
`year`	The year; an integer from *2000-2030*.

Response

Response Body
 	
(JSON)
	{
	  "date": "<*date*>",
	}

where:

Element	Description
`date`	System date in the following format: HH:MM:SS xM ZZZ Wkd Mon Dy YEAR where: • HH - hour • MM - minutes • SS - seconds • xM - one of "`AM`", "`PM`" • ZZZ - name of the time zone • Wkd - three-letter weekday abbreviation • Mon - three-letter month abbreviation • Dy - one or two-digit day • YEAR - four-digit year For example: `10:55:58 AM UTC Mon Jul  4 2016`

Set Clock Format

Sets the system clock format to 12 hour or 24 hour format.

Request

Method Type	PUT
Request URI	/nos/api/cfg/clock/format/

Request Body
 	
(JSON)
	{
	  "format": <*format*>,
	}

where:

Element	Description
`format`	System clock format; one of: • *12* (12 hour format) • *24* (24 hour format)

Response

Response Body
 	
(JSON)
	{
	  "format": <*format*>,
	}

where:

Element	Description
`format`	System clock format; one of: • *12* (12 hour format) • *24* (24 hour format)

Set Clock Protocol

Sets the clock protocol to either manual or Network Time Protocol (NTP).

Request

Method Type	PUT
Request URI	/nos/api/cfg/clock/protocol/

Request Body
 	
(JSON)
	{
	  "protocol": "*<protocol>*",
	}

where:

Element	Description
`protocol`	System clock protocol; one of: • *none* - the clock is manually configured • *ntp* - the clock is configured through NTP Default value: *ntp*.

Response

Response Body
 	
(JSON)
	{
	  "protocol": "*<protocol>*",
	}

where:

Element	Description
`protocol`	System clock protocol; one of: • *none* - the clock is manually configured • *ntp* - the clock is configured through NTP Default value: *ntp*.

Set Clock Timezone

Sets the clock time zone for the switch.

Request

Method Type	PUT
Request URI	/nos/api/cfg/clock/timezone

Request Body
 	
(JSON)
	{
	  "timezone": "*<timezone>*",
	  "offsethour": "*<offsethour>*",
	  "offsetmin": "*<lag_mode>*",
	}

where:

Element	Description
`timezone`	One to five letter string denoting the local system time zone.
`offsethour`	Hours offset from UTC; an integer from *-23* through *23*.
`offsetmin`	Minutes offset from UTC; an integer from *0-59*.

Response

Response Body
 	
(JSON)
	{
	  "date": "*<date>*",
	}

where:

Element	Description
`date`	System date in the following format: HH:MM:SS xM ZZZ Wkd Mon Dy YEAR where: • HH - hour • MM - minutes • SS - seconds • xM - one of "`AM`", "`PM`" • ZZZ - name of the time zone • Wkd - three-letter weekday abbreviation • Mon - three-letter month abbreviation • Dy - one or two-digit day • YEAR - four-digit year For example: `10:55:58 AM UTC Mon Jul  4 2016`

Set Clock Summer Time

Sets the transition to and from a summer time zone adjustment.

Request

Method Type	PUT
Request URI	/nos/api/cfg/clock/summertime

Request Body
 	
(JSON)
	{
	"timezone": *<time_zone>*,
	"startweek": *<start_week>*,
	"startweekday": *<start_weekday>*,
	"startmonth": *<start_month>*,
	"starttime" : "*<HH:MM>*",
	"endweek" : *<end_week>*,
	"endweekday": *<end_weekday>*,
	"endmonth" : *<end_month>*,
	"endtime" : "*<HH:MM>*",
	"offsetmin" : *<minutes>*
	}

where:

Element	Description
`timezone`	Local time zone of the system; a three to five character string such as *PST, MST, CST*, or *EST*.
`startweek`	Week number in the month in which to start Daylight Saving time; an integer from *1-5* (first week=1, last week=5).
`startweekday`	Weekday on which to start DST; one of the following case-insensitive strings: • *monday* • *tuesday* • *wednesday* • *thursday* • *friday* • *saturday* • *sunday*

Element	Description
`startmonth`	Month to start DST; one of the following case-insensitive strings: • *january* • *february* • *march* • *april* • *may* • *june* • *july* • *august* • *september* • *october* • *november* • *december*
`starttime`	Time to start DST; a string in the following format: HH:MM.
`endweek`	Week number in which to end DST; an integer from *1-5* (first week=1, last week=5).
`endweekday`	Weekday on which to end DST; one of the following case-insensitive strings: • *monday* • *tuesday* • *wednesday* • *thursday* • *friday* • *saturday* • *sunday*
`endmonth`	Month in which DST ends; one of the following case-insensitive strings: • *january* • *february* • *march* • *april* • *may* • *june* • *july* • *august* • *september* • *october* • *november* • *december*
`endtime`	Time to end DST; a string in the following format: HH:MM.
`offsetmin`	Offset to add, in minutes; an integer from *1-1440*.

Response

Response Body
 	
(JSON)
	{
	  "date": "<*date*>",
	}

where:

Element	Description
`date`	System date in the following format: HH:MM:SS xM ZZZ Wkd Mon Dy YEAR where: • HH - hour • MM - minutes • SS - seconds • xM - one of "`AM`", "`PM`" • ZZZ - name of the time zone • Wkd - three-letter weekday abbreviation • Mon - three-letter month abbreviation • Dy - one or two-digit day • YEAR - four-digit year For example: `10:55:58 AM UTC Mon Jul  4 2016`

Get Device Contact

Gets the device contact.

Request

Method Type	GET
Request URI	/nos/api/cfg/contact

Request Body
 	
(JSON)
	{
	  "contact": *<contact>*,
	}

where:

Element	Description
`contact`	Device contact; a string up to 256 characters long.

Response

Response Body
 	
(JSON)
	{
	  "contact": *<contact>*,
	}

where:

Element	Description
`contact`	Device contact; a string up to 256 characters long.

Update Device Contact

Updates the device contact.

Request

Method Type	PUT
Request URI	/nos/api/cfg/contact

Request Body
 	
(JSON)
	{
	  "contact": *<contact>*,
	}

where:

Element	Description
`contact`	Device contact; a string up to 256 characters long.

Response

Response Body
 	
(JSON)
	{
	  "contact": *<contact>*,
	}

where:

Element	Description
`contact`	Device contact; a string up to 256 characters long.

Get Device Description

Gets the device description.

Request

Method Type	GET
Request URI	/nos/api/cfg/descr

Request Body
 	
(JSON)
	{
	  "descr": *<descr>*,
	}

where:

Element	Description
`descr`	Device description; a string up to 256 characters long.

Response

Response Body
 	
(JSON)
	{
	  "descr": *<descr>*,
	}

where:

Element	Description
`descr`	Device description; a string up to 256 characters long.

Update Device Description

Updates the device description.

Request

Method Type	PUT
Request URI	/nos/api/cfg/descr

Request Body
 	
(JSON)
	{
	  "descr": *<descr>*,
	}

where:

Element	Description
`descr`	Device description; a string up to 256 characters long.

Response

Response Body
 	
(JSON)
	{
	  "descr": *<descr>*,
	}

where:

Element	Description
`descr`	Device description; a string up to 256 characters long.

Get Rack Properties

Gets the rack properties for the switch.

Request

Method Type	GET
Request URI	/nos/api/cfg/rack_prop
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"location":"*<location>*",
	"room":"*<room>*",
	"rack":"*<rack>*",
	"lru":"*<lru>*"
	}

where:

Element	Description
`location`	Device location; a string up to 256 characters long.
`room`	Device room ID; a string up to 256 characters long.
`rack`	Device Rack; a string up to 256 characters long.
`lru`	Device lowest rack unit; a string up to 256 characters long.

Update Rack Properties

Updates the rack properties for the switch.

Request

Method Type	PUT
Request URI	/nos/api/cfg/rack_prop

Request Body
 	
(JSON)
	{
	"location":"*<location>*",
	"room":"*<room>*",
	"rack":"*<rack>*",
	"lru":"*<lru>*"
	}

where:

Element	Description
`location`	(Optional) Device location; a string up to 256 characters long.
`room`	(Optional) Device room ID; a string up to 256 characters long.
`rack`	(Optional) Device Rack; a string up to 256 characters long.
`lru`	(Optional) Device lowest rack unit; a string up to 256 characters long.

Response

Response Body
 	
(JSON)
	{
	"location":"*<location>*",
	"room":"*<room>*",
	"rack":"*<rack>*",
	"lru":"*<lru>*"
	}

where:

Element	Description
`location`	Device location; a string up to 256 characters long.
`room`	Device room ID; a string up to 256 characters long.
`rack`	Device Rack; a string up to 256 characters long.
`lru`	Device lowest rack unit; a string up to 256 characters long.

System Information

The following system information URIs are available:

- /nos/api/sysinfo GET
- /nos/api/sysinfo/<*fans*> GET
- /nos/api/sysinfo/<*power*> GET
- /nos/api/sysinfo/<*temperature*> GET
- /nos/api/sysinfo/<*inventory*> GET
- /nos/api/sysinfo/serial_number GET
- /nos/api/sysinfo/panic_dump GET
- /nos/api/sysinfo/<*globalhealthstatus*> GET
- /nos/api/sysinfo/resources GET

The following system information commands are available:

- Get All System Information
- Get System Fan Information
- Get System Power Information
- Get System Temperature Information
- Get System Inventory
- Get System Serial Number
- Get Panic Dump Information
- Get Global Health Status
- Get System Resource Statistics

Get All System Information

Gets information about the system hardware.

Request

Request URI	/nos/api/sysinfo
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	"Fans":
	    {
	        “Fan 1”:
	        {
	            “Module” : “1”
	            “Air-flow” : ” Front-to-Back”,
	            “Speed-percent” : “0”,
	            “Speed-rpm” : “4205”
	        },
	        “Fan 2”:
	        {
	            “Module” : “1”
	            “Air-flow” : ” Front-to-Back”,
	            “Speed-percent” : “24”,
	            “Speed-rpm” : “4402”
	        }
	    },</pre>

	“Power”:
	{
	“power1”:
	{
	“Name” : “Power Supply 1”
	“Manufacturer” : ” DELTA”,
	“Model” : “XXXXXXXXXX”,
	“State” : “Normal ON”
	},
	“power2”:
	{
	“Name” : “Power Supply 2”
	“Manufacturer” : ” DELTA”,
	“Model” : “XXXXXXXXXX”,
	“State” : “12V Output Fault”
	}
	},
	“Temperature” :
	{
	“cpu”:
	{
	“Temp” : “31”
	“State” : ”OK”
	},
	“Ambient”:
	{
	“Temp” : “3O”,
	“State” : ”OK”
	},
	“Hot Spot” :
	{
	“Temp” : “46”,
	“State”: “OK”
	}
	“Temperature threshold” :
	{
	“System Warning” : 85,
	“System Shutdown” : 95,
	“System Set Point” : 7O
	}
	},
	“Inventory” :
	{
	“Name” : “8272”,
	“Description” : “G8272 (48x1OGE + 6x4OGE)”,
	“Model” : “LENOVO G8272”,
	“Manufacture Date”: “1452”,
	“Serial Number” : “YO52MV4CTOOJ”,
	“PCB Assembly” : “OOCJO67”,
	“Electronic Serial Number”: “MMO1267”,
	“Firmware Revision” : “O.O.O.O”,
	“Software Revision” : “O.O.O.O”,
	“Uuid” : “A48CDB33B6OOYO52MV4CTOOJ”,
	“Last reset Reason” : “Reset by CLI reload command”,
	"Service Led" : "enabled"
	},

	<pre>“Panic Dump”:
	{
	“File 3” : {
	“Name” : “nsm.gz”,
	“Date” : ”2016-05-31 22:38:03”
	},
	“File 2” : {
	“Name” : “hsl.gz”,
	“Date” : “2016-05-31 22:38:10”
	},
	“File 1” : {
	“Name” : ”imish.gz”,
	“Date” : ”2016-05-31 22:38:34”
	}
	}
	}</pre>

where:

Element	Description
`Fans`	System fan information.
`Powers`	System power information.
`Temperature`	System temperature information.
`Inventory`	System inventory.
`Panic Dump`	Panic dump information.

Get System Fan Information

Gets information about the system fans.

Request

Request URI	/nos/api/sysinfo/fans
Request Body (JSON)	

Response

Response Body
 	
(JSON)y
	{
	“Fan 1”:
	{
	“Module” : “1”
	“Air-flow” : ” Front-to-Back”,
	“Speed-percent” : “0”,
	“Speed-rpm” : “4205”
	},
	“Fan 2”:
	{
	“Module” : “1”
	“Air-flow” : ” Front-to-Back”,
	“Speed-percent” : “24”,
	“Speed-rpm” : “4402”
	}
	}

where:

Element	Description
`Module`	Module number.
`Air-flow`	Air flow type.
`Speed-percent`	Speed percentage.
`Speed-rpm`	Speed in RPM.

Get System Power Information

Gets information about the system power supplies.

Request

Request URI	/nos/api/sysinfo/power
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	  “power1”:
	  {
	  “Name” : “Power Supply 1”
	  “Manufacturer” : ” DELTA”,
	  “Model” : “XXXXXXXXXX”,
	  “State” : “Normal ON”
	  },
	  “power2”:
	  {
	  “Name” : “Power Supply 2”
	  “Manufacturer” : ” DELTA”,
	  “Model” : “XXXXXXXXXX”,
	  “State” : “12V Output Fault”
	  }
	},</pre>

where:

Element	Description
`name`	Power supply name.
`manufacturer`	Power supply manufacturer.
`model`	Power supply model.
`state`	Power supply state.

Get System Temperature Information

Gets information about the system temperature.

Request

Request URI	/nos/api/sysinfo/temperatures
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	“Cpu Local”:
	{
	“Temp” : “31”
	“State” : ”OK”
	},
	“Ambient”:
	{
	“Temp” : “30”,
	“State” : ”OK”
	},
	“Hot Spot” :
	{
	“Temp” : “46”,
	“State”: “OK”
	}
	“Temperature threshold” :
	{
	“System Warning” : 85,
	“System Shutdown” : 95,
	“System Set Point” : 70
	}
	},</pre>

where:

Element	Description
`temp`	The temperature.
`state`	The state.
`System warning`	Temperature at which a system warning is issued.
`System shutdown`	The temperature at which the system shuts down
`System set point`	The system set point temperature.

Get System Inventory

Gets information about the system inventory.

Request

Request URI	/nos/api/sysinfo/inventory
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	  “Name” : “8272”,
	  “Description” : “G8272 (48x10GE + 6x40GE)”,
	  “Model” : “LENOVO G8272”,
	  “Manufacture Date”: “1452”,
	  “Serial Number” : “Y052MV4CT00J”,
	  “PCB Assembly” : “00CJ067”,
	  “Electronic Serial Number” : “MM01267”,
	  “Firmware Revision” : “0.0.0.0”,
	  “Software Revision” : “0.0.0.0”,
	  “Uuid” : “A48CDB33B600Y052MV4CT00J”,
	  “Last reset Reason” : “Reset by CLI reload command”,
	  "Service Led" : "enabled"
	},

where:

Element	Description
`name`	System name.
`description`	System description.
`model`	System model.
`Manufacture Date`	System Manufacture Date.
`Serial Number`	System Serial Number.
`PCB Assembly`	System PCB Assembly.
`Electronic Serial Number`	System Electronic Serial Number.
`Firmware Revision`	System Firmware Revision.
`Software Revision`	System Software Revision.
`Uuid`	System UUID.
`Last reset Reason`	System last reset reason.
`Service Led`	Whether or not the Service LED is enabled.

Get System Serial Number

Gets the system serial number.

Request

Request URI	/nos/api/sysinfo/serial_number
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	  “Serial Number” : “YO52MV4CTOOJ”,
	}</pre>

where:

Element	Description
`Serial Number`	System Serial Number.

Get Panic Dump Information

Gets information about system panic dumps.

Request

Request URI	/nos/api/sysinfo/panic_dump
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	    {
	    “File 3” : {
	        “Name” : “nsm.gz”,
	        “Date” : ”2016-05-31 22:38:03”
	    },
	    “File 2” : {
	        “Name” : “hsl.gz”,
	        “Date” : “2016-05-31 22:38:10”
	    },
	    “File 1” : {
	        “Name” : ”imish.gz”,
	        “Date” : ”2016-05-31 22:38:34”
	    }
	}</pre>

where:

Element	Description
`Name`	File name.
`Date`	Date and time when the file was created.

Get Global Health Status

Gets information about system global health.

Request

Request URI	/nos/api/sysinfo/globalhealthstatus
Request Body (JSON)	

Response

Response Body (JSON)	{
	"status": "<*status*>",
	"description": "<*description*>"
	}

where:

Element	Description
`status`	System global health status; one of: • *OK* • *Noncritical* • *Critical*
`description`	Detailed description of the status; one of: • *OK*: o All temperature sensors are below the warning threshold; o All fans are running at >= 100 RPMs; o All power supplies are on; o No panic dump exists in flash. • *Noncritical*: o One or more temperature sensors is in the warning range; o A panic dump exists in flash. • *Critical*: o One or more temperature sensors is in the failure range; o One or more fans are running < 100 RPM; o One power supply is off.

Get System Resource Statistics

Gets system resource statistics, such as CPU statistics, memory statistics, tasks and load average.

Request

Method Type	GET
Request URI	/nos/api/sysinfo/resources
Request Body (JSON)	

Response

Response Body (JSON)


```
{
    "Cpu(s)":
    {
        "idle":"88.4",
        "hardware_interrupt":"0.0",
        "stolen_time":"0.0",
        "software_interrupt":"0.0",
        "io_wait":"0.1",
        "system":"2.9",
        "user_nice":"0.0",
        "user_un_nice":"8.7"
    },
    "Mem":
    {
        "total":"4000796",
        "buffers":"355260",
        "free":"3159012",
        "used":"486524"
    },
    "tasks":
    {
        "zombie":"0",
        "running":"1",
        "total":"108",
        "stopped":"0",
        "sleeping":"107"
    },
    "load average":
    {
        "5 min":"0.36",
        "15 min":"0.35",
        "1 min":"0.26"
    }
}
```

Telemetry

The following telemetry configuration-related URIs are available:

- /nos/api/info/telemetry/switch-properties GET
- /nos/api/cfg/telemetry/feature GET, PUT
- /nos/api/cfg/telemetry/bst/tracking GET, PUT
- /nos/api/cfg/telemetry/bst/feature GET, PUT
- /nos/api/info/telemetry/bst/report POST
- /nos/api/info/telemetry/bst/congestion-drop-counters POST
- /nos/api/cfg/telemetry/bst/threshold PUT, POST
- /nos/api/cfg/telemetry/bst/limits GET
- /nos/api/cfg/telemetry/bst/clear/threshold GET
- /nos/api/cfg/telemetry/bst/clear/statistics GET
- /nos/api/cfg/telemetry/clear-cgsn-drop-counters GET
- nos/api/cfg/telemetry/cancel-request PUT
- /nos/api/info/statistics/interface/{*if_name*} GET
- /nos/api/cfg/statistics/clear/interface/{*if_name*} DELETE

The following telemetry configuration commands are available:

- Get Switch Properties
- Set System Feature
- Get System Feature
- Set BST Tracking
- Get BST Tracking
- Set BST Feature
- Get BST Feature
- Get BST Limits
- Get BST Report
- Get BST Congestion Drop Counters
- Set BST Threshold
- Get BST Threshold
- Clear BST Threshold
- Clear BST Statistics
- Clear BST Congestion Drops
- Cancel BST Request

- Get Interface Statistics
- Clear Interface Statistics

Get Switch Properties

Gets system switch properties.

Request

Method Type	GET
Request URI	/nos/api/info/telemetry/switch-properties
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>"time-stamp": "2015-10-18 - 00:15:04",
	  { "number-of-asics": 1,
	        "asic-info": [["1",
	"BCM56850", 78]],
	       "supported-features": ["BST"],
	        "network-os": "CNOS",
	        "uid": "0000d80bb99bbbbb",
	        "agent-ip": "192.168.1.2",
	        "agent-port": "8080",
	        "agent-sw-version": "3.0.0.1"
	}</pre>

where:

Element	Description
`number-of-asics`	Number of asics in the switch; integer format.
`asic-info`	List of dictionaries; one of: • `asic-id`: ASIC identifier; string • `chip-id`: part number of the silicon; string • `num-ports`: Number of ports available on the switch and managed by this ASIC; an integer
`supported features`	A list of strings indicating the features supported by the Agent.
`network-os`	The Network Operating system currently used on the switch.
`uid`	Unique identifier for this switch. This unique ID is the key for the SDN controller to map the switch to the nodes existing in their discovery database.
`agent-ip`	IP address of the switch where the Agent is running; string

Element	Description
`agent-port`	TCP port number of the switch, at which the Agent is listening; string
`agent-sw-version`	Software version number for the Agent; string

Set System Feature

Sets system feature.

Request

Method Type	PUT
Request URI	/nos/api/cfg/telemetry/feature

Request Body
 	
(JSON)
	<pre>{
	  "heartbeat-enable" : 1,
	  "msg-interval" : 10
	}</pre>

where:

Element	Description
`heartbeat-enable`	When enabled, the Agent asynchronously sends the registration and heartbeat message to the collector. One of: • 0: disable heartbeat • 1: enable heartbeat (default value)
`msg-interval`	Determines the interval with which the registration and heartbeat messages are sent to the collector; units of seconds from 1-600. Default value: 5 seconds.

Response

Response Body (JSON)	

Get System Feature

Gets system feature.

Request

Method Type	GET
Request URI	/nos/api/cfg/telemetry/feature
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	  "heartbeat-enable" : 1,
	  "msg-interval" : 5
	}</pre>

where:

Element	Description
`heartbeat-enable`	When enabled, the Agent asynchronously sends the registration and heartbeat message to the collector. One of: • 0: disable heartbeat • 1: enable heartbeat (default value)
`msg-interval`	Determines the interval with which the registration and heartbeat messages are sent to the collector; units of seconds from 1-600. Default value: 5 seconds.

Set BST Tracking

Sets the BST trackers and the tracking-mode on the ASIC.

Request

Method Type	PUT
Request URI	/nos/api/cfg/telemetry/bst/tracking

Request Body
 	
(JSON)
	{
	"track-peak-stats”:1,
	"track-ingress-port-priority-group":1,
	"track-ingress-port-service-pool":1,
	"track-ingress-service-pool":1,
	"track-egress-port-service-pool":1,
	"track-egress-service-pool":1,
	"track-egress-rqe-queue":1,
	“track-egress-cpu-queue”:1,
	“track-egress-uc-queue”:1,
	“track-egress-mc-queue”:1,
	"track-device":1
	}

where:

Element	Description
`track-peak-stats`	Set to 1 to peak statistics tracking, 0 to disable this feature
`track-ingress-port-priority-group`	Set to 1 to enable ingress port priority group tracking, 0 to disable this feature
`track-ingress-port-service-pool`	Set to 1 to enable ingress port service pool tracking, 0 to disable this feature
`track-ingress-service-pool`	Set to 1 to enable ingress service pool tracking, 0 to disable this feature
`track-egress-port-service-pool`	Set to 1 to enable egress port service pool tracking, 0 to disable this feature
`track-egress-service-pool`	Set to 1 to enable egress service pool tracking, 0 to disable this feature
`track-egress-rqe-queue`	Set to 1 to enable egress RQE queue tracking, 0 to disable this feature
`track-egress-cpu-queue`	Set to 1 to enable egress CPU queue tracking, 0 to disable this feature
`track-egress-uc-queue`	Set to 1 to enable egress unicast queue tracking, 0 to disable this feature

Element	Description
`track-egress-mc-queue`	Set to 1 to enable egress multicast queue tracking, 0 to disable this feature
`track-device`	Set to 1 to enable tracking of this device, 0 to disable this feature

Response

Response Body (JSON)	

Get BST Tracking

Gets the BST trackers and the tracking-mode on the ASIC.

Request

Method Type	GET
Request URI	/nos/api/cfg/telemetry/bst/tracking
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"track-peak-stats":1,
	"track-ingress-port-priority-group":1,
	"track-ingress-port-service-pool":1,
	"track-ingress-service-pool":1,
	"track-egress-port-service-pool":1,
	"track-egress-service-pool":1,
	"track-egress-rqe-queue":1,
	“track-egress-cpu-queue”:1,
	“track-egress-uc-queue”:1,
	“track-egress-mc-queue”:1,
	"track-device":1
	}

where:

Element	Description
`track-peak-stats`	1 to peak statistics tracking, 0 to disable this feature
`track-ingress-port-priority-group`	1 to enable ingress port priority group tracking, 0 to disable this feature
`track-ingress-port-service-pool`	1 to enable ingress port service pool tracking, 0 to disable this feature
`track-ingress-service-pool`	1 to enable ingress service pool tracking, 0 to disable this feature
`track-egress-port-service-pool`	1 to enable egress port service pool tracking, 0 to disable this feature
`track-egress-service-pool`	1 to enable egress service pool tracking, 0 to disable this feature
`track-egress-rqe-queue`	1 to enable egress RQE queue tracking, 0 to disable this feature

Element	Description
`track-egress-cpu-queue`	Set to 1 to enable egress CPU queue tracking, 0 to disable this feature
`track-egress-uc-queue`	Set to 1 to enable egress unicast queue tracking, 0 to disable this feature
`track-egress-mc-queue`	Set to 1 to enable egress multicast queue tracking, 0 to disable this feature
`track-device`	1 to enable tracking of this device, 0 to disable this feature

Set BST Feature

Sets BST feature.

Request

Method Type	PUT
Request URI	/nos/api/cfg/telemetry/bst/feature

Request Body
 	
(JSON)
	<pre>{
	"bst-enable": 1,
	"send-async-reports": 1,
	"collection-interval": 300,
	"trigger-rate-limit": 5,
	"trigger-rate-limit-interval": 2,
	"send-snapshot-on-trigger": 1,
	"async-full-reports": 1,
	}</pre>

where:

Element	Description
`bst-enable`	Set to 1 to enable BST, 0 to disable it. Enabling BST allows the switch to track buffer utilization statistics.
`send-async-reports`	Set to 1 to enable the transmission of periodic asynchronous reports, 0 to disable this feature.
`collection-interval`	The collection interval, in seconds. This defines how frequently periodic reports will be sent to the configured controller; an integer from *10 - 600*.
`trigger-rate-limit`	The trigger rate limit, which defines the maximum number of threshold-driven triggered reports that the agent is allowed to send to the controller per `trigger-rate-limit-interval`; an integer from 1-5.
`trigger-rate-limit-interval`	The trigger rate limit interval, in seconds; an integer from 10-60.
`send-snapshot-on-trigger`	Set to 1 to enable sending a complete snapshot of all buffer statistics counters when a trigger happens, 0 to disable this feature.
`async-full-report`	Set to 1 to enable the async full report feature, 0 to disable it. When this feature is enabled, the agent sends full reports containing data related to all counters. When the feature is disabled, the agent sends incremental reports containing only the counters that have changed since the last report.

Response

Response Body (JSON)	

Get BST Feature

Gets BST information.

Request

Method Type	GET
Request URI	/nos/api/cfg/telemetry/bst/feature
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	    "stat-units-in-cells": 0,
	    "stats-in-percentage": 1,
	    "collection-interval": 60,
	    "send-async-reports": 0,
	    "send-snapshot-on-trigger": 1,
	    "trigger-rate-limit": 1,
	    "async-full-report": 0,
	    "trigger-rate-limit-interval": 10,
	    "bst-enable": 0
	}</pre>

where:

Element	Description
`bst-enable`	Set to 1 to enable BST, 0 to disable it. Enabling BST allows the switch to track buffer utilization statistics.
`send-async-reports`	Set to 1 to enable the transmission of periodic asynchronous reports, 0 to disable this feature.
`collection-interval`	The collection interval, in seconds. This defines how frequently periodic reports will be sent to the configured controller.
`trigger-rate-limit`	The trigger rate limit, which defines the maximum number of threshold-driven triggered reports that the agent is allowed to send to the controller per `trigger-rate-limit-interval`; an integer from 1-5.
`trigger-rate-limit-interval`	The trigger rate limit interval, in seconds; an integer from 10-60.
`send-snapshot-on-trigger`	Set to 1 to enable sending a complete snapshot of all buffer statistics counters when a trigger happens, 0 to disable this feature.

Element	Description
`async-full-report`	Set to 1 to enable the async full report feature, 0 to disable it. When this feature is enabled, the agent sends full reports containing data related to all counters. When the feature is disabled, the agent sends incremental reports containing only the counters that have changed since the last report.
`stat-units-in-cells`	Whether the buffer statistics are reported in units of bytes or cells. **Note:** This value is always set to 0. It cannot be modified and it is always ignored because `stats-in-percentage` is always set to 1.
`stats-in-percentage`	When set to 1, the buffer usage statistics are reported as percentages. When this variable is set to 1, the parameter `stat-units-in-cells` is ignored while reporting the statistics. This variable is applicable for statistics and threshold reporting. **Note:** This variable is always set to 1 and cannot be modified. The percentage values in the BST/trigger report are an approximation of buffer utilization, not an exact value.

Get BST Limits

Gets BST limit information for BST parameters.

Request

Method Type	GET
Request URI	nos/api/cfg/telemetry/bst/limits
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>"time-stamp": "2015-10-18 - 00:15:04 ",
	{
	    "cpu-queue": {"max": 47, "min": 0},
	    "multicast-queue": {"max": 2599, "min": 0},
	    "priority-group": {"max": 7, "min": 7},
	    "queue-group": {"max": 127, "min": 0},
	    "rqe-queue": {"max": 10, "min": 0},
	    "service-pool": {"max": 0, "min": 0},
	    "unicast-queue": {"max": 2599, "min": 0},
	    "user-queue": {"max": 7, "min": 0}
	}</pre>

The range of the following parameters depends on the hardware configuration. For example, the service-pool and priority-group ranges are different when Converged Ethernet Mode is enabled on the switch:

Element	Description
`cpu-queue`	Cpu-queue minimum and maximum values.
`multicast-queue`	Multicast-queue minimum and maximum values.
`priority-group`	Priority-group minimum and maximum values.
`queue-group`	Queue-group minimum and maximum values.
`rqe-queue`	Rqe-queue minimum and maximum values.
`service-pool`	Service-pool minimum and maximum values.
`unicast-queue`	Unicast-queue minimum and maximum values.
`user-queue`	User-queue minimum and maximum values.

Get BST Report

Gets BST information.

Request

Method Type	POST
Request URI	/nos/api/info/telemetry/bst/report

Request Body
 	
(JSON)
	{
	"include-ingress-port-priority-group" : 1,
	"include-ingress-port-service-pool" : 0,
	"include-ingress-service-pool" : 0,
	"include-egress-port-service-pool" : 0,
	"include-egress-service-pool" : 1,
	"include-egress-rqe-queue" : 0,
	"include-egress-uc-queue" : 1,
	"include-egress-mc-queue" : 1,
	"include-egress-cpu-queue": 0,
	"include-device" : 0
	}

where:

Element	Description
`include-ingress-port-priority-group`	Ingress port priority group; 1 to enable, 0 to disable.
`include-ingress-port-service-pool`	Ingress port service pool; 1 to enable, 0 to disable.
`include-ingress-service-pool`	Ingress service pool; 1 to enable, 0 to disable.
`include-egress-port-service-pool`	Egress port service pool; 1 to enable, 0 to disable.
`include-egress-service-pool`	Egress service pool; 1 to enable, 0 to disable.
`include-egress-rqe-queue`	Egress RQE queue; 1 to enable, 0 to disable.
`include-egress-uc-queue`	Egress unicast queue buffers. Set to 1 to enable in BST report, 0 to disable it.
`include-egress-cpu-queue`	Egress CPU queue buffers. Set to 1 to enable in BST report, 0 to disable it.
`include-egress-mc-queue`	Egress multicast queue buffers. Set to 1 to enable in BST report, 0 to disable it.
`include-device`	Device; 1 to enable, 0 to disable.

Response

Response Body
 	
(JSON)
	{
	
	{ "time-stamp": "2014-11-14 - 00:15:04 ",
	"report": [
	{ "realm": "device",
	"data": "46"
	},
	{ "realm": "ingress-port-priority-group",
	"data": [
	{ "interface": " Ethernet1/2",
	"data": [[5, "100", "100"]]
	}, { "interface": " Ethernet1/3",
	"data": [[5, "100", "100"]]
	}]
	},
	{ "realm": "ingress-port-service-pool",
	"data": [
	{ "interface": "Ethernet1/2",
	"data": [[5, "100"]]
	}, { "interface ": "Ethernet1/3",
	"data": [[6, "100"]]
	}]
	},
	{ "realm": "ingress-service-pool",
	"data": [[1, "100"], [2, "100"]]
	},
	{ "realm": "egress-cpu-queue",
	"data": [[3, "100"]]
	},
	{ "realm": "egress-uc-queue",
	"data": [[3, "100"]]
	},
	{ "realm": "egress-mc-queue",
	"data": [[3, "100"]]
	},
	{ "realm": "egress-port-service-pool",
	" port-service-pool-ctr": [{
	"interface": " Ethernet1/2",
	"data": [["5","10", "10", "30"]]
	}, { "interface": " Ethernet1/3",
	"data": [["60", "30", "36","45"]]
	}]
	},
	{ "realm": "egress-rqe-queue",
	"data": [[2, "33"], [5, "25"]]
	},
	{ "realm": "egress-service-pool",
	"data": [[1,"20", "10", "10", "32"],
	[3, "3660", 0, 0]]
	},
	]

where:

Realm	Index # 1	Index # 2	Statistics
ingress-port-priority-group	*interface* (such as Ethernet1/7)	priority-group	um-share-buffer-count um-head room-buffer-count
ingress-port-service-pool	*interface* (such as Ethernet1/7)	service-pool	um-share-buffer-count
ingress-service-pool	service-pool		um-share-buffer-count
egress-port-service-pool	*interface* (such as Ethernet1/7)	service-pool	uc-share-buffer-count, um-share-buffer-count, mc-share-buffer-count,
egress-cpu-queue	queue		cpu-buffer-count
egress-uc-queue	queue		uc-buffer-count
egress-mc-queue	queue		mc-buffer-count
egress-service-pool	service-pool		um-share-buffercount, mc-share-buffer-count
egress-rqe-queue	queue		rqe-buffer-count
device			data

Note: For more information on realm parameters and indexes, see the *CNOS Application Guide*.

Get BST Congestion Drop Counters

Gets BST congestion drop counters information.

Request

Method Type	POST
Request URI	/nos/api/info/telemetry/bst/congestion-drop-counters

Request Body
 	
(JSON)
	{
	req_id : 1
	"request-type" : "top-drops" or "top-port-queue-drops" or
	"port-drops" or " port-queue-drops"
	"request-params": {
	"count":8
	"interface-list" :["if_name1", "if_name2", "if_name3"]
	"queue-type" : "ucast" or "mcast" or "all"
	"queue-list" : [1, 2, 3]
	},
	"collection-interval": 30
	}

where:

Element	Description
`req-id`	The request ID; an integer
`request-type`	One of the following: • `top-drops`: Show ports with maximum congestion on the switch and their drop-counters • `top-port-queue-drops`: Show top port-queue level drop-counters on the switch • `port-queue-drops`: Show per port-queue level drop-counters on the switch • `port-drops`: Show per-port total drop counters on the switch

Element	Description
`request-params`	Request parameters; one of the following strings: • `count`: Number of ports required in the report. The ports are sorted with the port suffering maximum congestion at the top; an integer. **Note:** This parameter is configurable only if `request-type` is `top-drops` or `top-port-queue-drops` • `queue-type`: Filters the report on the queue type; one of the following strings: – `ucast`: Unicast queues – `mcast`: Multicast queues – `all`: All supported queues • `interface-list`: Comma-separated list of ports for the congestion drop counter report; an array. A value of all requests all the ports. • `queue-list`: An array of queue numbers to be considered for the drop report. • `collection-interval`: (Optional) The period in which the counters are collected from ASIC; An integer from 1-60. Default value: 0

Response

Method Type	POST
Request URI	/nos/api/info/telemetry/bst/congestion-drop-counters

Request Body
 	
(JSON)
	{
	"time-stamp": "2017-01-02 - 14:54:22 ",
	"report-type": "port-drops",
	"congestion-ctr": [
	{"interface": "Ethernet1/1", "ctr": "56776 "},
	{"interface": "Ethernet1/2", "ctr": "56767"},
	{"interface": "Ethernet1/3", "ctr": " 76654"}
	]
	}

where:

Element	Description
`time-stamp`	Time of the report generation.

Element	Description
`report-type`	One of the following: • `top-drops`: Show ports with maximum congestion on the switch and their drop-counters • `top-port-queue-drops`: Show top port-queue level drop-counters on the switch • port-queue-drops: Show per port-queue level drop-counters on the switch • `port-drops`: Show per-port total drop counters on the switch
`congestion-ctr`	Congestion counters contents; a list of dictionaries. Depending on the configuration, each dictionary may contain the following values: • `interface`: Interface name • `ctr`: Counter value; a string. • `queue-type`; one of ucast, mcast • `queue-drop-ctr`; one of: – `queue number`: an integer from 1-8. – `counter value`: the 64 bit counter value; a string.

Port Drop Report Example

Request

Method Type	POST
Request URI	/nos/api/info/telemetry/bst/congestion-drop-counters

Request Body
 	
(JSON)
	{
	"req_id " : 1
	"request-type" : "port-drops"
	"request-params":
	{
	"interface-list" : ["Ethernet1/1", "Ethernet1/2",
	"Ethernet1/3"]
	}

Response

Response Body
 	
(JSON)
	{
	"time-stamp": "2017-01-02 - 14:54:22 ",
	"report-type": "port-drops",
	"congestion-ctr": [
	{"interface": "Ethernet1/1", "ctr": "56776 "},
	{"interface": "Ethernet1/2", "ctr": "56767"},
	{"interface": "Ethernet1/3", "ctr": " 76654"}
	]
	}

Top Drop Report Example

Request

Method Type	POST
Request URI	/nos/api/info/telemetry/bst/congestion-drop-counters

Request Body
 	
(JSON)
	{
	"req-id" : 2,
	"request-type" : "top-drops",
	"request-params": {
	"count":3
	}
	}

Response

Response Body
 	
(JSON)
	{
	"time-stamp": "2017-01-02 - 14:54:22 ",
	"report-type": "top-drops",
	"congestion-ctr": [
	{"interface": "Ethernet1/1", "ctr": " 1234 "},
	{"interface": "Ethernet1/2", "ctr": " 3234"},
	{"interface": "Ethernet1/3", "ctr": " 3455"}
	]
	}

Port Queue Drops Report Example

Request

Method Type	POST
Request URI	/nos/api/info/telemetry/bst/congestion-drop-counters

Request Body
 	
(JSON)
	{
	"req-id" : 4,
	"request-type" : "port-queue-drops",
	"request-params": {
	"interface-list": ["Ethernet1/1", "Ethernet1/2"],
	"queue-type": "mcast",
	"queue-list" : [1,2] }
	}

Response

Response Body
 	
(JSON)
	{
	"time-stamp": "2017-01-02 - 14:40:01 ",
	"report-type": "port-queue-drops",
	"congestion-ctr": [
	{
	"interface": "Ethernet1/1",
	"queue-type": "mcast",
	"queue-drop-ctr": [[1, "0 "], [2, " 0 "]]
	},
	{
	"interface": "Ethernet1/2",
	"queue-type": "mcast",
	"queue-drop-ctr": [[1, "0 "], [2, " 0 "]]
	}
	]
	}

Top Port Queue Drops Report Example

Request

Method Type	POST
Request URI	/nos/api/info/telemetry/bst/congestion-drop-counters

Request Body
 	
(JSON)
	{
	"req-id" : 3,
	"request-type" : "top-port-queue-drops",
	"request-params":
	{
	"count":5 ,
	"queue-type": "ucast"
	}
	}

Response

Response Body
 	
(JSON)
	{
	{
	"time-stamp": "2017-01-02 - 14:43:39 ",
	"report-type": "top-port-queue-drops",
	"congestion-ctr": [
	{
	"interface": "Ethernet1/1",
	"queue-type": "ucast",
	"queue-drop-ctr": [[1, "0"], [2, "0"], [3, "0"], [4,"0"],
	[5, "0"]]
	}
	]
	}

Set BST Threshold

Sets BST threshold to trigger BST reports. Use the following REST APIs to set thresholds for each realm.

Request

Method Type	PUT
Request URI	nos/api/cfg/telemetry/bst/threshold

Request Body
(JSON)
	{
	"realm": "ingress-service-pool",
	"service-pool": 0,
	"um-share-threshold": 70
	}

where:

Realm	Index # 1	Index # 2	Thresholds
ingress-port-priority-group	*interface* (such as Ethernet1/7)	priority-group	um-share-threshold
ingress-port-service-pool	*interface* (such as Ethernet1/7)	service-pool	um-share-threshold
ingress-service-pool	service-pool		um-share-threshold
egress-port-service-pool	service-pool		uc-share-threshold, um-share-threshold
egress-service-pool	service-pool		um-share-threshold mc-share-threshold
egress-rqe-queue	queue		rqe-threshold
egress-cpu-queue	queue		cpu-threshold
egress-uc-queue	queue		uc-threshold
	interface	user-queue	uc-threshold
egress-mc-queue	queue		mc-threshold
	interface	user-queue	mc-threshold
include-device			threshold

Note: For more information on realm parameters and indexes, see the *CNOS Application Guide*.

Response

Response Body (JSON)	

Get BST Threshold

Retrieves BST threshold.

Request

Method Type	POST
Request URI	nos/api/cfg/telemetry/bst/threshold

Request Body
 	
(JSON)
	{
	"include-ingress-port-priority-group" : 1,
	"include-ingress-port-service-pool" : 1,
	"include-ingress-service-pool" : 1,
	"include-egress-port-service-pool" : 1,
	"include-egress-service-pool" : 1,
	“include-egress-cpu-queue”:1,
	“include-egress-uc-queue”:1,
	“include-egress-mc-queue”:1
	}

where:

Element	Description
`include-ingress-port-priority-group`	Ingress port priority group; 1 to enable, 0 to disable.
`include-ingress-port-service-pool`	Ingress port service pool; 1 to enable, 0 to disable.
`include-ingress-service-pool`	Ingress service pool; 1 to enable, 0 to disable.
`include-egress-port-service-pool`	Egress port service pool; 1 to enable, 0 to disable.
`include-egress-service-pool`	Egress service pool; 1 to enable, 0 to disable.
`include-egress-cpu-queue`	Egress CPU queue buffers. Set to 1 to enable in BST threshold report, 0 to disable it.
`include-egress-uc-queue`	Egress unicast queue buffers. Set to 1 to enable in BST threshold report, 0 to disable it.
`include-egress-mc-queue`	Egress multicast queue buffers. Set to 1 to enable in BST threshold report, 0 to disable it.

Response

Response Body

(JSON)


```
{
    "report": [
        {
            "realm": "ingress-port-service-pool",
            "data": [
                {
                    "interface": "Ethernet1/40",
                    "data": [
                        [
                            0,
                            "100"
                        ]
                    ]
                },
                {
                    "interface": "Ethernet1/1",
                    "data": [
                        [
                            0,
                            "100"
                        ]
                    ]
                },
...
        {
            "realm": "ingress-service-pool",
            "data": [
                [
                    0,
                    "100"
                ]
            ]
        }
...
```


where:

Realm	Index # 1	Index # 2	Thresholds
ingress-port-priority-group	*interface* (such as Ethernet1/7)	priority-group	um-share-threshold um-head room-threshold
ingress-port-service-pool	*interface* (such as Ethernet1/7)	service-pool	um-share-threshold
ingress-service-pool	service-pool		um-share-threshold
egress-port-service-pool	*interface* (such as Ethernet1/7)	service-pool	uc-share-threshold, um-share-threshold mc-share-threshold
egress-service-pool	service-pool		um-share-threshold mc-share-threshold
egress-rqe-queue	queue		rqe-threshold
egress-cpu-queue	queue		cpu-threshold

Realm	Index # 1	Index # 2	Thresholds
egress-uc-queue	queue		uc-threshold
egress-mc-queue	queue		mc-threshold
device			threshold

Note: For more information on realm parameters and indexes, see the *CNOS Application Guide*.

Clear BST Threshold

Clears BST threshold.

Request

Method Type	GET
Request URI	nos/api/cfg/telemetry/bst/clear/threshold
Request Body (JSON)	

Response

Response Body (JSON)	

Clear BST Statistics

Clears BST statistics.

Request

Method Type	GET
Request URI	nos/api/cfg/telemetry/bst/clear/statistics
Request Body (JSON)	

Response

Response Body (JSON)	

Clear BST Congestion Drops

Clears BST congestion drop.

Request

Method Type	GET
Request URI	nos/api/cfg/telemetry/clear-cgsn-drop-counters
Request Body (JSON)	

Response

Response Body (JSON)	

Cancel BST Request

Cancels BST request.

Request

Method Type	PUT
Request URI	nos/api/cfg/telemetry/cancel-request

Request Body (JSON)	<pre>{
	  "req-id":200,
	  "cancel-req-id":1
	}</pre>

where:

Element	Description
`req-id`	The unique request ID; an integer from *1-10,000.*
`cancel-req-id`	The request ID of the periodic report to cancel; an integer from *1-10,000.*

Response

On success, a HTTP 200 OK response is sent with no JSON body.

Response Body (JSON)	

Get Interface Statistics

Gets statistics for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/info/statistics/interface<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Response

Following is a command response example:

Response Body (JSON)	{
	"tx_pkts_65_to_127_bytes": "0",
	"rx_mcast_pkts": "0",
	"rx_pkts": "0",
	"tx_dropped": "0",
	"if_name": "Ethernet1/1",
	"rx_pkts_65_to_127_bytes": "0",
	"rx_rate": "0",
	"tx_oversize_pkts": "0",
	"tx_pkts_512_to_1023_bytes": "0",
	"rx_pkts_256_to_511_bytes": "0",
	"if_down_drops": "0",
	"rx_pkts_1519_to_1548_bytesg": "0",
	"tx_bytes": "0",
	"rx_ucast_pkts": "0",
	"rx_crc_errors": "0",
	"tx_mcast_pkts": "0",
	"tx_rate": "0",
	"tx_ucast_pkts": "0",
	"rx_pkts_1024_to_1518_bytes": "0",
	"tx_pkts_128_to_255_bytes": "0",
	"rx_pkts_1519_to_1548_bytes": "0",
	"rx_pkts_0_to_64_bytes": "0",
	"rx_bcast_pkts": "0",
	"rx_bytes": "0",
	"tx_errors": "0",
	"tx_pkts_0_to_64_bytes": "0",
	"tx_bcast_pkts": "0",
	"rx_bytes": "0",
	"tx_errors": "0",
	"tx_pkts_0_to_64_bytes": "0",
	"tx_bcast_pkts": "0",
	"rx_pause": "0",
	"tx_pkts_1024_to_1518_bytes": "0",
	"rx_undersize_pkts": "0",
	"tx_pkts_256_to_511_bytes": "0",
	"rx_input_discards": "0",
	"tx_pause": "0",
	"rx_pkts_128_to_255_bytes": "0",
	"rx_errors": "0",
	"tx_pkts": "0",
	"rx_pkts_512_to_1023_bytes": "0",
	"tx_bit_rate": "0",
	"rx_bit_rate": "0",
	"tx_pkts_1519_to_1548_bytes": "0",
	"rx_oversize_pkts": "0"
	}

Clear Interface Statistics

Resets the statistics for a specific switch interface.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/statistics/clear/interface/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	The name of the switch interface. For example: *Ethernet1/12*.

Telnet

The following Telnet URI is available:

- /nos/api/cfg/telnet/server GET, PUT

The following Telnet commands are available:

- Get Telnet Server
- Set Telnet Server

Get Telnet Server

Gets the Telnet server status.

Request

Method Type	GET
Request URI	/nos/api/cfg/telnet/server
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	    "status": "{enable | disable}"
	}</pre>

where:

Element	Description
`status`	The Telnet server status (string); one of *enable, disable*.

Set Telnet Server

Sets the Telnet server status.

Request

Method Type	PUT
Request URI	/nos/api/cfg/telnet/server

Request Body
 	
(JSON)
	<pre>{
	    "status": "{enable/disable}"
	}</pre>

where:

Element	Description
`status`	The Telnet server status (string); one of *enable, disable*.

Response

Response Body
 	
(JSON)
	<pre>{
	    "status": "{enable | disable}"
	}</pre>

where:

Element	Description
`status`	The Telnet server status (string); one of *enable, disable*.

TACACS+

The following Terminal Access Controller Access-Control System Plus (TACACS+) URIs are available:

- /nos/api/cfg/tacacs GET, PUT
- /nos/api/cfg/tacacs/hosts GET, POST, DELETE
- /nos/api/cfg/tacacs/groups GET, POST, DELETE

The following TACACS+ commands are available:

- Get TACACS+ Configuration
- Update TACACS+ Configuration
- Get TACACS+ Server Configuration
- Add TACACS+ Server
- Delete TACACS+ Server
- Get TACACS+ Server Group Configuration
- Add TACACS+ Server Group
- Delete TACACS+ Server Group

Get TACACS+ Configuration

Gets the TACACS+ configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/tacacs
Request Body (JSON)	

Response

Response Body (JSON)	`{`
	`  "status":"{enable|disable}",`
	`  "global_key":"{configured|not configured}"`
	`}`

where:

Element	Description
`status`	The status of the TACACS+ service on the switch; one of *enable, disable.*
`global_key`	The status of the global TACACS+ encryption/decryption key; one of *configured, not configured.*

Update TACACS+ Configuration

Updates the TACACS+ configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/tacacs

Request Body (JSON)	{
	"status":"{enable|disable}",
	"global_key":"{*global_key*}",
	"global_key_form":{0|7}
	}

where:

Element	Description
`status`	The status of the TACACS+ service on the switch; one of *enable, disable.*
`global_key`	The global TACACS+ encryption/decryption key; a string up to 63 characters long.
`global_key_form`	The encryption method for the global TACACS+ key; one of *0* (clear text), *7* (encrypted).

Response

Response Body (JSON)	{
	"status":"{enable|disable}",
	"global_key":"{configured|not configured}"
	}

where:

Element	Description
`status`	The status of the TACACS+ service on the switch; one of *enable, disable.*
`global_key`	The status of the global TACACS+ encryption/decryption key; one of *configured, not configured.*

Get TACACS+ Server Configuration

Gets the configuration of a specific configured TACACS+ server or of all configured TACACS+ servers.

Request

Method Type	GET
Request URI	/nos/api/cfg/tacacs/hosts/<*IP_addr*>
Request Body (JSON)	

where:

Element	Description
IP_addr	The IP address of the configured TACACS+ server. If no IP address if provided, then the command returns the configuration of all configured TACACS+ servers.

Response

Response Body (JSON)	`[`
	`  {`
	`  "IP_addr":"{`*IP_addr*`}",`
	`  "port":{`*port*`},`
	`  "key":"{configured|not configured}"`
	`  }`
	`]`

where:

Element	Description
`IP_addr`	The IP address of the configured TACACS+ server.
`port`	The TCP port used to connect to the TACACS+ server; an integer from *1-65535*.
`key`	The status of the TACACS+ server encryption/decryption key; one of *configured, not configured*.

Add TACACS+ Server

Configures a TACACS+ server.

Request

Method Type	POST
Request URI	/nos/api/cfg/tacacs/hosts

Request Body (JSON)	{
	"IP_addr":"{*IP_addr*}",
	"port":{*port*},
	"key":"{*key*}",
	"key_form":{0|7}
	}

where:

Element	Description
IP_addr	The IP address of the TACACS+ server.
port	The TCP port used to connect to the TACACS+ server; an integer from *1-65535.*
key	The TACACS+ server encryption/decryption key; a string up to 63 characters long.
key_form	The encryption method for the TACACS+ server key; one of *0* (clear text), *7* (encrypted).

Response

Response Body (JSON)	{
	"IP_addr":"{*IP_addr*}",
	"port":{*port*},
	"key":"{configured|not configured}"
	}

where:

Element	Description
IP_addr	The IP address of the TACACS+ server.
port	The TCP port used to connect to the TACACS+ server; an integer from *1-65535.*
key	The status of the TACACS+ server encryption/decryption key; one of *configured, not configured.*

Delete TACACS+ Server

Removes a configured TACASC+ server.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/tacacs/hosts/<*IP_addr*>
Request Body (JSON)	

where:

Element	Description
IP_addr	The IP address of the TACACS+ server.

Get TACACS+ Server Group Configuration

Gets the configuration of a specific configured TACACS+ server group or of all configured TACACS+ server groups.

Request

Method Type	GET
Request URI	/nos/api/cfg/tacacs/groups/<*group_name*>
Request Body (JSON)	

where:

Element	Description
group_name	The name of the TACACS+ server group; a string up to 127 characters long. **Note:** If no group name is provided, then the command returns the configuration of all configured TACACS+ server groups.

Response

Response Body (JSON)

```
[
   {
      "group_name":"{group_name}"
      "vrf_name":"{vrf_name}",
      "hosts":
         [
            {
               "IP_addr":"{IP_addr}",
               "port":{port},
               "key":"{configured|not configured}"
            }
         ]
   }
]
```

where:

Element	Description
`group_name`	The name of the TACACS+ server group; a string up to 127 characters long.
`vrf_name`	The VRF instance for the TACACS+ server group. Valid value: the *VRF instance name.*
`hosts`	The list of servers members of the TACACS+ server group.
`IP_addr`	The IP address of the TACACS+ server.

Element	Description
`port`	The TCP port used to connect to the TACACS+ server; an integer from *1-65535.*
`key`	The status of the TACACS+ server encryption/decryption key; one of *configured, not configured.*

Add TACACS+ Server Group

Configures a TACACS+ server group.

Request

Method Type	POST
Request URI	/nos/api/cfg/tacacs/groups

Request Body (JSON)	{
	"group_name":"{group_name}",
	"vrf_name":"{vrf_name}",
	"hosts":
	[
	"{*IP_addr*}"
	]
	}

where:

Element	Description
`group_name`	The name of the TACACS+ server group; a string up to 127 characters long.
`vrf_name`	The VRF instance for the TACACS+ server group. Valid value: the*VRF instance name.*
`hosts`	The list of servers members of the TACACS+ server group.
IP_addr	The IP address of the TACACS+ server to be added to the group.

Response

Response Body (JSON)	[
	  {
	  "group_name":"{*group_name*}"
	  "vrf_name":"{*vrf_name*}",
	  "hosts":
	  [
	  {
	  "IP_addr":"{*IP_addr*}",
	  "port":{*port*},
	  "key":"{configured|not configured}"
	  }
	  ]
	  }
	]

where:

Element	Description
`group_name`	The name of the TACACS+ server group; a string up to 127 characters long.
`vrf_name`	The VRF instance for the TACACS+ server group. Valid value: the *VRF instance name.*
`hosts`	The list of servers members of the TACACS+ server group.
`IP_addr`	The IP address of the TACACS+ server.
`port`	The TCP port used to connect to the TACACS+ server.; an integer from *1-65535.*
`key`	The status of the TACACS+ server encryption/decryption key; one of *configured, not configured.*

Delete TACACS+ Server Group

Removes a configured TACACS+ server group.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/tacacs/groups/<*group_name*>
Request Body (JSON)	

where:

Element	Description
group_name	The name of the TACACS+ server group; a string up to 127characters long.

vLAG

The following Virtual Link Aggregation Group (vLAG) URIs are available:

- /nos/api/cfg/vlag GET, PUT
- /nos/api/info/vlag GET
- /nos/api/info/vlag/isl GET
- /nos/api/cfg/vlag/isl PUT
- /nos/api/info/vlag/health_check GET
- /nos/api/cfg/vlag/health_check PUT
- /nos/api/cfg/vlag/instance POST
- /nos/api/cfg/vlag/instance/<*instance_id*> GET, PUT, DELETE
- /nos/api/info/vlag/instance/<*instance_id*> GET

The following vLAG commands are available:

- Get vLAG Configuration
- Update vLAG Configuration
- Get Global vLAG Information
- Get vLAG ISL Information
- Configure vLAG ISL
- Get vLAG Health Check Information
- Configure vLAG Health Check Parameters
- Create vLAG Instance
- Update vLAG Instance
- Delete vLAG Instance
- Get vLAG Instance Configuration
- Get vLAG Instance Information

Get vLAG Configuration

Gets the Virtual Link Aggregation Group (vLAG) global configuration.

Request

Method Type	GET
Request URI	/nos/api/cfg/vlag
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"status": "*<status>*",
	"tier_id": "*<tier_id>*",
	"priority": "*<priority>*",
	"auto_recover" : "*<auto_recover>*",
	"startup_delay": "*<startup_delay>*",
	}

where:

Element	Description
`status`	Whether the vLAG is enabled or disabled; one of *enable, disable.* Default value: *disable*.
`tier_id`	vLAG tier ID value; an intger from *1-512.* Default value: *0*.
`priority`	vLAG priority value; an integer from *0-65535*. Default value: *0*.
`auto_recover`	Time interval, in seconds; an integer from *240-3600*. Default value: *300*.
`startup_delay`	Delay time, in seconds; an integer from *0-3600.* Default value: *120*.

Update vLAG Configuration

Updates the Virtual Link Aggregation Group (vLAG) global configuration.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vlag

Request Body
 	
(JSON)
	{
	"status": "*<status>*",
	"tier_id": "*<tier_id>*",
	"priority": "*<priority>*",
	"auto_recover" : "*<auto_recover>*",
	"startup_delay": "*<startup_delay>*",
	}

where:

Element	Description
`status`	Whether the vLAG is enabled or disabled; one of *enable, disable.* Default value: *disable*.
`tier_id`	vLAG tier ID value; an intger from *1-512.* Default value: *0*.
`priority`	vLAG priority value; an integer from *0-65535*. Default value: *0*.
`auto_recover`	Time interval, in seconds; an integer from *240-3600*. Default value: *300*.
`startup_delay`	Delay time, in seconds; an integer from *0-3600.* Default value: *120*.

Note: At least one parameter must be specified in the request body.

Response

Response Body
 	
(JSON)
	{
	"status": "*<status>*",
	"tier_id": "*<tier_id>*",
	"priority": "*<priority>*",
	"auto_recover" : "*<auto_recover>*",
	"startup_delay": "*<startup_delay>*",
	}

where:

Element	Description
`status`	Whether the vLAG is enabled or disabled; one of *enable, disable.* Default value: *disable*.
`tier_id`	vLAG tier ID value; an intger from *1-512.* Default value: *0*.

Element	Description
`priority`	vLAG priority value; an integer from *0-65535*. Default value: *0*.
`auto_recover`	Time interval, in seconds; an integer from *240-3600*. Default value: *300*.
`startup_delay`	Delay time, in seconds; an integer from *0-3600*. Default value: *120*.

Get Global vLAG Information

Gets global vLAG information.

Request

Method Type	GET
Request URI	/nos/api/info/vlag
Request Body (JSON)	

Response

Response Body (JSON)	{ "status": "*<status>*", "system_mac": "*<system_mac>*", "fdb_refresh": "*<fdb_refresh>*", "fdb_synch" : "*<fdb_synch>*", "auto_recovery": { "interval": "*<interval>*" "state": "*<state>*", } "startup_delay": { "interval": "*<interval>*" "state": "*<state>*", } "local": { “tier_id”: "*<tier_id>*", “sys_type”: "*<sys_type>*", “os_version”: "*<os_version>*", “admin_role”: "*<admin_role>*", “oper_role”: "*<oper_role>*", “priority” : "*<priority>*", “system_mac”: "*<system_mac>*", “match”: "*<match>*" } "peer": { “tier_id”: "*<tier_id>*", “sys_type”: "*<sys_type>*", “os_version”: "*<os_version>*", “admin_role”: "*<admin_role>*", “oper_role”: "*<oper_role>*", “priority” : "*<priority>*", “system_mac”: "*<system_mac>*", “match”: "*<match>*" } }

where:

Element	Description
`status`	Whether the vLAG is enabled or disabled; one of *enable, disable*. Default value: *disable*.
`system_mac`	Unique vLAG system MAC used for LACP negotiation on the vLAG ports so the access switch forms a single LAG. The vLAG *tier_id* is used to form this vLAG system MAC.
`fdb_ refresh`	Whether FDB refresh is configured; one of *yes, no*.
`fdb_synch`	Whether FDB is synchronized; one of *yes, no*.
`auto_ recovery`	A dictionary consisting of the following values: • *interval*: Time interval, in seconds; an integer from 240-3600. Default value: 300. • *state*: Auto-recovery state; one of `unstarted`, `running`, `finished`.
`startup_ delay`	A dictionary consisting of the following values: • *interval*: Delay time, in seconds; an integer from 0-3600. Default value: 120. • *state*: Startup delay state; one of `unstarted`, `running`, `finished`.
`local`	Dictionary containing the following values: • *tier_id*: vLAG tier ID of the local switch. • *sys_type*: Lenovo hardware model number. • *os_version* CNOS version. • *admin_role*: One of `Primary`, `Secondary`, `Unselected`. • *oper_role*: One of `Primary`, `Secondary`, `Unselected`, • *priority*: The local vLAG priority • *system_mac*: Local switch MAC. • *match*: Whether there is an ISL local match or mismatch; one of `Match`, `Mis-Match`.
`peer`	Dictionary containing the following values: • *tier_id*: vLAG tier ID of the peer switch. • *sys_type*: Lenovo hardware model number. • *os_version* CNOS version. • *admin_role*: One of `Primary`, `Secondary`, `Unselected`. • *oper_role*: One of `Primary`, `Secondary`, `Unselected`, • *priority*: The peer vLAG priority • *system_mac*: Peer switch MAC. • *match*: Whether there is an ISL local match or mismatch; one of `Match`, `Mis-Match`.

Get vLAG ISL Information

Gets Virtual Link Aggregation Group (vLAG) Inter-Switch Link (ISL) information.

Request

Method Type	GET
Request URI	/nos/api/info/vlag/isl
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"port_aggregator": "<*port_aggregator*>",
	"if_index": "<*if_index*>",
	"state": "<*state*>",
	"prev_state" : "<*prev_state*>",
	}

where:

Element	Description
`port_aggregator`	LAG identifier; an integer from *1-4096*.
`if_index`	ISL interface index.
`state`	ISL state; one of *Down, Inactive, Active*.
`prev_state`	Previous ISL state; one of *Down, Inactive, Active*.

Configure vLAG ISL

Configures the port aggregator for the vLAG ISL.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vlag/isl

Request Body
 	
(JSON)
	{
	"port_aggregator": *"<port_aggregator>"*,
	}

where:

Element	Description
`port_aggregator`	Port aggregator for the vLAG ISL.

Response

Response Body (JSON)	

Get vLAG Health Check Information

Gets vLAG health check information.

Request

Method Type	GET
Request URI	/nos/api/info/vlag/health_check
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	<pre>{
	  "status" : "<*status*>",
	  "peer_ip": "<*peer_ip*>",
	  "vrf": "<*vrf*>",
	  "local_ip": "<*local_ip*>",
	  "retry_interval": "<*retry_interval*>",
	  "keepalive_attempts" : "<*keepalive_attempts*>",
	  "keepalive_interval" : "<*keepalive_interval*>",
	}</pre>

where:

Element	Description
`status`	vLAG health check status; one of *up, down.*
`peer_ip`	IP address of peer switch. This can be the management IP address of the peer switch.
`vrf`	VRF context string.
`local_ip`	IP address of local switch. This can be the management IP address of the local switch.
`retry_interval`	Time interval, in seconds; an integer from *1-300*. Default value: *30*.
`keepalive_attempts`	Number of keepalive attempts made before declaring the peer is down; an integer from *1-24*. Default value: *3*.
`keepalive_interval`	Time interval, in seconds; an integer from *2-300*. Default value: *5*.

Configure vLAG Health Check Parameters

Configures vLAG health check parameters.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vlag/health_check

Request Body
 	
(JSON)
	{
	"peer_ip": "*<peer_ip>*",
	"vrf": "*<vrf>*",
	"retry_interval": "*<retry_interval>*",
	"keepalive_attempts" : "*<keepalive_attempts>*",
	"keepalive_interval" : "*<keepalive_interval>*",
	}

where:

Element	Description
peer_ip	IP address of peer switch. This can be the management IP address of the peer switch.
vrf	VRF context string.
retry_interval	Time interval, in seconds; an integer from *1-300*. Default value: *30*.
keepalive_attempts	Number of keepalive attempts made before declaring the peer is down; an integer from *1-24*. Default value: *3*.
keepalive_interval	Time interval, in seconds; an integer from *2-300*. Default value: *5*.

Response

Response Body
 	
(JSON)
	{
	"status" : "<status>",
	"peer_ip": "<peer_ip>",
	"vrf": "<vrf>",
	"local_ip": "<local_ip>",
	"retry_interval": "<retry_interval>",
	"keepalive_attempts" : "<keepalive_attempts>",
	"keepalive_interval" : "<keepalive_interval>",
	}

where:

Element	Description
peer_ip	IP address of peer switch. This can be the management IP address of the peer switch.
vrf	VRF context string.

Element	Description
`retry_interval`	Time interval, in seconds; an integer from *1-300*. Default value: *30*.
`keepalive_attempts`	Number of keepalive attempts made before declaring the peer is down; an integer from *1-24*. Default value: *3*.
`keepalive_interval`	Time interval, in seconds; an integer from *2-300*. Default value: *5*.

Create vLAG Instance

Creates a Virtual Link Aggregation Group (vLAG) instance.

Request

Method Type	POST
Request URI	/nos/api/cfg/vlag/instance

Request Body
 	
(JSON)
	{
	"inst_id": "*<inst_id>*",
	"port_aggregator": "*<port_aggregator>*",
	"status": "*<status>*",
	}

where:

Element	Description
`inst_id`	vLAG instance ID number; an integer from *1-64*.
`port_ aggregator`	LAG identifier; an integer from *1-4096*.
`status`	vLAG status; one of *enable, disable*. Default value: *disable*.

Response

Response Body (JSON)	

Update vLAG Instance

Updates a Virtual Link Aggregation Group (vLAG) instance.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vlag/instance/<*instance_id*>

Request Body
 	
(JSON)
	{
	"port_aggregator": "<*port_aggregator*>",
	"status": "<*status*>",
	}

where:

Element	Description
instance_id	vLAG instance ID number; an integer from *1-64*.
`port_aggregator`	LAG identifier; an integer from *1-4096*.
`status`	vLAG status; one of *enable, disable*. Default value: *disable*.

Response

Response Body (JSON)	

Delete vLAG Instance

Deletes a vLAG instance.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/vlag/instance/<*instance_id*>
Request Body (JSON)	

where:

Element	Description
instance_id	vLAG instance ID number; an integer from 1-64.

Response

`True` if the operation succeeded; otherwise `False`.

Get vLAG Instance Configuration

Gets configuration parameters for the specified vLAG instance.

Note: An *instance_id* value of `None` returns configuration parameters for all vLAG instances.

Request

Method Type	GET
Request URI	/nos/api/cfg/vlag/instance/<*instance_id*>
Request Body (JSON)	

where:

Element	Description
instance_id	vLAG instance ID; either `None` or an integer from 1-64.

Response

Response Body
 	
(JSON)
	<pre>{
	  "port_aggregator": 6,
	  "status": "enable",
	  "inst_id": 2
	}</pre>

where:

Element	Description
`inst_id`	vLAG instance ID number; an integer from *1-64*.
`port_aggregator`	LAG identifier; an integer from *1-4096*.
`status`	vLAG status; one of *enable, disable*. Default value: *disable*.

Get vLAG Instance Information

Gets information about a vLAG instance.

Note: An *instance_id* value of `None` returns information about all vLAG instances.

Request

Method Type	GET
Request URI	/nos/api/info/vlag/instance/<*instance_id*>
Request Body (JSON)	

where:

Element	Description
instance_id	vLAG instance ID; either `None` or an integer from 1-64.

Response

Response Body
 	
(JSON)
	{
	"port_aggregator": "<*port_aggregator*>",
	"inst_id": "<*inst_id*>",
	"state": "<*state*>",
	"prev_state" : "<*prev_state*>",
	}

where:

Element	Description
`port_aggregator`	LAG identifier; an integer from *1-4096*.
`inst_id`	ISL interface index.
`state`	ISL state; one of *Down, Inactive, Active*.
`prev_state`	Previous ISL state; one of *Down, Inactive, Active*.

VLAN

The following VLAN-related URIs are available:

- /nos/api/cfg/vlan GET, POST
- /nos/api/cfg/vlan/<*vlan_id*> GET, PUT, DELETE
- /nos/api/cfg/vlan_interface GET
- /nos/api/info/subnetvlan POST
- /nos/api/cfg/subnetvlan POST, PUT
- /nos/api/cfg/vlan_interface/<*if_name*> GET, PUT

The following VLAN commands are available:

- Get All VLANs
- Create VLAN
- Get VLAN
- Update VLAN
- Delete VLAN
- Get VLAN Properties of All Interfaces
- Get VLAN Interface Properties
- Update VLAN Interface Properties

Get All VLANs

Gets properties of all VLANs.

Request

Method Type	GET
Request URI	/nos/api/cfg/vlan/
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	{
	"vlan_name": "*<vlan_name>*",
	"vlan_id": "*<vlan_id>*",
	"admin_state": "*<admin_state>*",
	"mst_inst_id": "*<mst_inst_id>*",
	"interfaces": [
	{ "if_name": "*<if_name>*",
	"bridgeport_mode": "*<bridgeport_mode>*",
	"pvid": "*<pvid>*"
	}
	]
	}
	]

where:

Element	Description
`vlan_name`	The name of the VLAN.
`vlan_id`	VLAN number.; an integer from *2-3999*.
`admin_state`	The admin status; one of *up, down*.
`mst_inst_id`	MST instance ID; an integer from *0-64*. Default value: *0*. **Note:** Instance 0 refers to the CIST.
`interfaces`	Interface members of a VLAN. **Note:** The interface members must exist.
`if_name`	Ethernet interface name (string). **Note:** The ethernet interface must exist.
`bridgeport_mode`	Bridge port mode; one of *access, trunk*.
`pvid`	Native VLAN for a port (he access VLAN for access ports or the native VLAN for trunk ports); an integer from *1-3999*. Default value: *1*.

Create VLAN

Creates a VLAN.

Request

Method Type	POST
Request URI	/nos/api/cfg/vlan

Request Body
 	
(JSON)
	{
	"vlan_name": "*<vlan_name>*",
	"vlan_id": "*<vlan_id>*",
	"admin_state": "*<admin_state>*",
	}

where:

Element	Description
vlan_name	VLAN name; a string up to 32 characters long. To create a VLAN with the default name, the *vlan_name* field must be null.
vlan_id	VLAN number.; an integer from *2-3999*.
admin_state	The admin status; one of *up, down.*

Response

Response Body
 	
(JSON)
	{
	"vlan_name": "Vlan10",
	"interfaces": [],
	"admin_state": "up",
	"vlan_id": 10,
	"mst_inst_id": 1
	}

where:

Element	Description
vlan_name	The name of the VLAN.
vlan_id	VLAN number.; an integer from *2-3999*.
admin_state	The admin status; one of *up, down.*

Get VLAN

Gets properties of a VLAN.

Request

Method Type	GET
Request URI	/nos/api/cfg/vlan/*<vlan_id>*
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"vlan_name": "*<vlan_name>*",
	"vlan_id": "*<vlan_id>*",
	"admin_state": "*<admin_state>*",
	"mst_inst_id": "*<mst_inst_id>*",
	"interfaces": [
	{ "if_name": "*<if_name>*",
	"bridgeport_mode": "*<bridgeport_mode>*",
	"pvid": "*<pvid>*"
	}
	]
	}

where:

Element	Description
vlan_name	The name of the VLAN.
vlan_id	VLAN number.; an integer from *2-3999*.
admin_state	The admin status; one of *up, down*.
mst_inst_id	MST instance ID; an integer from *0-64*. Default value: *0*. **Note:** Instance 0 refers to the CIST.
interfaces	Interface members of a VLAN. **Note:** The interface members must exist.
if_name	Ethernet interface name (string). **Note:** The ethernet interface must exist.
bridgeport_mode	Bridge port mode; one of *access, trunk*.
pvid	Native VLAN for a port (he access VLAN for access ports or the native VLAN for trunk ports); an integer from *1-3999.* Default value: *1*.

Update VLAN

Updates properties of a VLAN.

Note: If an element is not specified in a PUT request, no update for that element will be performed.

Request

Method Type	PUT
Request URI	`/nos/api/cfg/vlan/`*<vlan_id>*

Request Body
 	
(JSON)
	`{`
	`  "vlan_name": "`*<vlan_name>*`",`
	`  "admin_state": "`*<admin_state>*`",`
	`}`

where:

Element	Description
`vlan_name`	VLAN name; a string up to 32 characters long. To change a VLAN name with default name, the *vlan_name* field must be null.
`admin_state`	The admin status; one of *up, down.*

Response

Response Body
 	
(JSON)
	`{`
	`"vlan_name": "`*<vlan_name>*`",`
	`"vlan_id": "`*<vlan_id>*`",`
	`"admin_state": "`*<admin_state>*`",`
	`"mst_inst_id": "`*<mst_inst_id>*`",`
	`"interfaces": [`
	`  {  "if_name": "`*<if_name>*`",`
	`  "bridgeport_mode": "`*<bridge_port_mode>*`",`
	`  "pvid": "`*<pvid>*`"`
	`  }`
	`]`
	`}`

where:

Element	Description
`vlan_name`	The name of the VLAN.
`vlan_id`	VLAN number.; an integer from *2-3999*.
`admin_state`	The admin status; one of *up, down*.
`mst_inst_id`	MST instance ID; an integer from *0-64*. Default value: *0*. **Note:** Instance 0 refers to the CIST.

Element	Description
`interfaces`	Interface members of a VLAN. **Note:** The interface members must exist.
`if_name`	Ethernet interface name (string). **Note:** The ethernet interface must exist.
`bridgeport_mode`	Bridge port mode; one of *access, trunk*.
`pvid`	Native VLAN for a port (he access VLAN for access ports or the native VLAN for trunk ports); an integer from *1-3999*. Default value: *1*.

Delete VLAN

Deletes a VLAN.

Note: If the specified *vlan_id* is `all`, all user-created VLANs will be deleted.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/vlan/<*vlan_id*>
Request Body (JSON)	

Get VLAN Properties of All Interfaces

Gets VLAN properties for all switch interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/vlan_interface
Request Body (JSON)	

Response

Response Body (JSON)	`[`
	`  {`
	`  "if_name": "{`*if_name*`}",`
	`  "bridgeport_mode": "{`*bridgeport_mode*`}",`
	`  "pvid": "{`*pvid*`}",`
	`  "vlans": ["{vlan_id}"],`
	`  "egress_type": "{`*egress_type*`}",`
	`  "egress_type_vlans" : ["{vlan_id}"]`
	`  }`
	`]`

where:

Element	Description
`if_name`	The name of the switch interface. For example *Ethernet1/12*.
`bridgeport_mode`	The bridge port mode; one of *access, trunk,* or *invalid* for routed port.
`pvid`	The Native VLAN ID for switch ports set up as trunk ports, or the Access VLAN ID for switch ports set up as access ports; an integer from *1 - 3999.* Default value: *1.*
`vlans`	The VLANs that the switch port is a member of; one of *all, none, 1 - 3999.*
`vlan_id`	The VLAN ID; an integer from *1- 3999.*
`egress_type`	Whether traffic is egress tagged when the interface is in Hybrid mode; one of *tagged, untagged.*
`egress_type_vlans`	The VLANs on which traffic is egress tagged.

Get VLAN Interface Properties

Gets VLAN properties for a specific switch interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/vlan_interface/<*if_name*>
Request Body (JSON)	

Response

Response Body (JSON)	{
	"if_name": "{*if_name*}",
	"bridgeport_mode": "{*bridgeport_mode*}",
	"pvid": "{*pvid*}",
	"vlans": ["{vlan_id}"],
	"egress_type": "{*egress_type*}",
	"egress_type_vlans" : ["{vlan_id}"]
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example: *Ethernet1/12*.
`bridgeport_mode`	The bridge port mode; one of *access, trunk.*
`pvid`	The Native VLAN ID for switch ports set up as trunk ports, or the Access VLAN ID for switch ports set up as access ports; an integer from *1 - 3999.* Default value: *1.*
`vlans`	The VLANs that the switch port is a member of; one of *all, none, 1 - 3999.*
`vlan_id`	The VLAN ID; an integer from *1- 3999.*
`egress_type`	Whether traffic is egress tagged when the interface is in Hybrid mode; one of *tagged, untagged.*
`egress_type_vlans`	The VLANs on which traffic is egress tagged.

Update VLAN Interface Properties

Updates the VLAN properties for a specific switch interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vlan_interface/<*if_name*>

Request Body (JSON)	{
	"if_name": "{*if_name*}",
	"bridgeport_mode": "{*bridgeport_mode*}",
	"pvid": "{*pvid*}",
	"vlans": ["{vlan_id}"],
	"egress_type": "{*egress_type*}",
	"egress_type_vlans" : ["{vlan_id}"]
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example *Ethernet1/12*.
`bridgeport_mode`	The bridge port mode; one of *access, trunk.*
`pvid`	The Native VLAN ID for switch ports set up as trunk ports, or the Access VLAN ID for switch ports set up as access ports; an integer from *1 - 3999.* Default value: *1.*
`vlans`	The VLANs that the switch port is a member of; one of *all, none, 1 - 3999.*
`vlan_id`	The VLAN ID; an integer from *1- 3999.*
`egress_type`	Whether traffic is egress tagged when the interface is in Hybrid mode; one of *tagged, untagged.*
`egress_type_vlans`	The VLANs on which traffic is egress tagged.

Response

Response Body (JSON)	{
	"if_name": "{*if_name*}",
	"bridgeport_mode": "{*bridgeport_mode*}",
	"pvid": "{*pvid*}",
	"vlans": ["{vlan_id}"],
	"egress_type": "{*egress_type*}",
	"egress_type_vlans" : ["{vlan_id}"]
	}

where:

Element	Description
`if_name`	The name of the switch interface. For example *Ethernet1/12*.
`bridgeport_mode`	The bridge port mode; one of *access, trunk.*
`pvid`	The Native VLAN ID for switch ports set up as trunk ports, or the Access VLAN ID for switch ports set up as access ports; an integer from *1 - 3999.* Default value: *1.*
`vlans`	The VLANs that the switch port is a member of; one of *all, none, 1 - 3999*
`vlan_id`	The VLAN ID; an integer from *1- 3999.*
`egress_type`	Whether traffic is egress tagged when the interface is in Hybrid mode; one of *tagged, untagged.*
`egress_type_vlans`	The VLANs on which traffic is egress tagged.

VLAN Interface Properties

The following VLAN interface property URI is available:

- /nos/api/cfg/vlan_interface GET, PUT
- /nos/api/cfg/vlan_interface/<*if_name*> GET, PUT

The following VLAN interface property commands are available:

- Get VLAN Properties of All Interfaces
- Get VLAN Interface Properties
- Update VLAN Interface Properties
- Update VLAN Interface Allowed VLAN List

Get VLAN Properties of All Interfaces

Gets the VLAN properties of all Ethernet interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/vlan_interface
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "if_name": "*<if_name>*",
	  "bridgeport_mode": "*<bridgeport_mode>*",
	  "pvid": "*<pvid>*",
	  "vlans": [
	  "*<vlan_id>*"
	  ]
	  }
	]

where:

Element	Description
`if_name`	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
`bridgeport_mode`	Bridge port mode; one of *access, trunk.*
`pvid`	Native VLAN for a port (he access VLAN for access ports or the native VLAN for trunk ports); an integer from *1-3999*. Default value: *1*.
`vlans`	VLAN memberships; either a*ll, none,* or an integer from *1-3999*.

Get VLAN Interface Properties

Gets the VLAN properties of an Ethernet interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/vlan_interface/<*if_name*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"bridgeport_mode": "<*bridge_port_mode*>",
	"pvid": "<*pvid*>",
	"vlans": ["<*vlan_id*>"]
	}

where:

Element	Description
if_name	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
`bridgeport_mode`	Bridge port mode; one of *access, trunk.*
`pvid`	Native VLAN for a port (he access VLAN for access ports or the native VLAN for trunk ports); an integer from *1-3999*. Default value: *1*.
`vlans`	VLAN memberships; either a*ll, none*, or an integer from *1-3999*.

Update VLAN Interface Properties

Updates the VLAN properties of an Ethernet interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vlan_interface/*<if_name>*

Request Body
 	
(JSON)
	{
	"if_name": "*<if_name>*",
	"bridgeport_mode": "*<bridgeport_mode>*"
	"pvid": "*<pvid>*",
	"vlans": ["*<vlan_id>*"]
	}

where:

Element	Description
if_name	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
bridgeport_mode	Bridge port mode; one of *access, trunk.*
pvid	Native VLAN for a port (he access VLAN for access ports or the native VLAN for trunk ports); an integer from *1-3999*. Default value: *1*.
vlans	VLAN memberships; either a*ll, none*, or an integer from *1-3999*.

Note: If an element is not specified in a PUT request, no update for that element will be performed.

Response

Response Body
 	
(JSON)
	{
	"if_name": "*<if_name>*",
	"bridge_port": "*<bridge_port>*",
	"bridgeport_mode": "*<bridgeport_mode>*",
	"pvid": "*<pvid>*",
	"vlans": ["*<vlans>*"]
	}

where:

Element	Description
if_name	Ethernet interface name (string). **Note:** The Ethernet interface must exist.
bridge_port	Whether or not the port is a bridge port; one of *yes, no*. Default value: *yes*.

Element	Description
`bridgeport_mode`	Bridge port mode; one of *access, trunk.*
`pvid`	Native VLAN for a port (the access VLAN for access ports or the native VLAN for trunk ports); an integer from *1-3999*. Default value: *1*.
`vlans`	(Optional) VLAN memberships; *all, none,* or an integer from *1-3999*.

Update VLAN Interface Allowed VLAN List

Updates the allowed VLAN list for a specific VLAN interface.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vlan_interface

Request Body (JSON)	{
	“if_name”:”{*if_name*}”,
	“bridgeport_mode”:”{access|trunk|hybrid}”,
	“pvid”:”{*pvid*}”,
	“vlans”:
	[
	“operation”:”{add|remove|except}”,
	“vlan_id”:”{*vlan_id*}”
	]
	“egress_type”:”{tagged|untagged}”,
	“egress_type_vlans”:”{*vlan_range*}”
	}

where:

Element	Description
`if_name`	The name of the switch interface.
`bridgeport_mode`	The bridge port mode; one of *access, trunk, hybrid.*
`pvid`	The access VLAN, if bridge port mode is configured to access. The native VLAN, if bridge port mode is configured to trunk; an integer from *1-3999.* Default value: *1.*
`vlans`	The list of allowed VLANs.
`operation`	The type of operation to perform; one of: • *add* - adds the VLAN to the allowed VLAN list • *remove* - removes the VLAN from the allowed VLAN list • *except* - excepts the VLAN from the allowed VLAN list
`vlan_id`	The VLANs to add, remove, or make an exception for; an integer from *1-3999.*
`egress_type`	Whether the switch tags egress traffic when in hybrid bridge port mode; one of *tagged, untagged.*
`egress_type_vlans`	The VLANs on which the switch tags egress traffic; an integer from *1-3999.*

Response

Response Body (JSON)	{
	“if_name”:”{*if_name*}”,
	“bridgeport_mode”:”{access|trunk|hybrid}”,
	“pvid”:”{*pvid*}”,
	“vlans”:
	[
	“operation”:”{add|remove|except}”,
	“vlan_id”:”{*vlan_id*}”
	]
	“egress_type”:”{tagged|untagged}”,
	“egress_type_vlans”:”{*vlan_range*}”
	}

where:

Element	Description
`if_name`	The name of the switch interface.
`bridgeport_mode`	The bridge port mode; one of *access, trunk, hybrid.*
`pvid`	The access VLAN, if bridge port mode is configured to access. The native VLAN, if bridge port mode is configured to trunk; an integer from *1-3999.* Default value: *1.*
`vlans`	The list of allowed VLANs.
`operation`	The type of operation to perform; one of: • *add* - adds the VLAN to the allowed VLAN list • *remove* - removes the VLAN from the allowed VLAN list • *except* - excepts the VLAN from the allowed VLAN list
`vlan_id`	The VLANs to add, remove, or make an exception for; an integer from *1-3999.*
`egress_type`	Whether the switch tags egress traffic when in hybrid bridge port mode; one of *tagged, untagged.*
`egress_type_vlans`	The VLANs on which the switch tags egress traffic; one of *1-3999.*

VRF

The following VRF URIs are available:

- /nos/api/cfg/vrf GET, POST
- /nos/api/cfg/vrf/<*vrf_name*> GET, PUT, DELETE

The following VRF commands are available:

- Get All VRFs
- Create VRF
- Get VRF
- Update VRF
- Delete VRF

Get All VRFs

Gets properties of all VRFs.

Request

Method Type	GET
Request URI	/nos/api/cfg/vrf
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "vrf_name": "*<vrf_name>*",
	  "interfaces": ["*<if_name>*"]
	  }
	]

where:

Element	Description
`vrf_name`	VRF name; a string up to 63 characters long.
`interfaces`	Interface members of the VRF. **Note:** The interfaces must exist.

Create VRF

Creates a new VRF.

Request

Method Type	POST
Request URI	/nos/api/cfg/vrf

Request Body
 	
(JSON)
	{
	  "vrf_name": "*<vrf_name>*",
	  "interfaces": [
	  "*<if_name>*"
	  ]
	}

where:

Element	Description
`vrf_name`	VRF name; a string up to 63 characters long.
`interfaces`	Interface members of the VRF. **Note:** The interfaces must exist.

Response

Response Body
 	
(JSON)
	{
	  "vrf_name": "*<vrf_name>*",
	  "interfaces": [
	  "*<if_name>*"
	  ]
	}

where:

Element	Description
`vrf_name`	VRF name; a string up to 63 characters long.
`interfaces`	Interface members of the VRF. **Note:** The interfaces must exist.

Get VRF

Gets properties of one VRF.

Request

Method Type	GET
Request URI	/nos/api/cfg/vrf/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

Response

Response Body
 	
(JSON)
	`[`
	`  {`
	`  "vrf_name": "`<*vrf_name*>`",`
	`  "interfaces": ["`<*if_name*>`"]`
	`  }`
	`]`

where:

Element	Description
`vrf_name`	VRF name; a string up to 63 characters long.
`interfaces`	Interface members of the VRF. **Note:** The interfaces must exist.

Update VRF

Updates the properties of a VRF.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vrf/<*vrf_name*>

Request Body
 	
(JSON)
	{
	"vrf_name": "<*vrf_name*>",
	"interfaces": [
	"<*if_name*>"
	]
	}

where:

Element	Description
`vrf_name`	VRF name; a string up to 63 characters long.
`interfaces`	Interface members of the VRF. **Note:** The interfaces must exist.

Response

Response Body
 	
(JSON)
	{
	"vrf_name": "<*vrf_name*>",
	"interfaces": [
	"<*if_name*>"
	]
	}

where:

Element	Description
`vrf_name`	VRF name; a string up to 63 characters long.
`interfaces`	Interface members of the VRF. **Note:** The interfaces must exist.

Delete VRF

Deletes a VRF.

Note: If the specified *vrf_name* is `all`, all user-created VRFs will be deleted.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/vrf/<*vrf_name*>
Request Body (JSON)	

where:

Element	Description
vrf_name	VRF name; one of the VRF name, `default`, `all`. Default value: `default`.

VRRP

The following VRRP URIs are available:

- /nos/api/cfg/vrrp GET
- /nos/api/cfg/vrrp/<*if_name*> GET, POST
- /nos/api/cfg/vrrp/<*if_name*>/<*vr_id*> GET, PUT, DELETE

The following VRRP commands are available:

- Get VRRP VRs of All Interfaces
- Get VRRP VRs of One Interface
- Create VRRP VR
- Get VRRP VR
- Update VRRP VR
- Delete VRRP VR

Get VRRP VRs of All Interfaces

Gets properties of all VRRP VRs of all interfaces.

Request

Method Type	GET
Request URI	/nos/api/cfg/vrrp
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	[
	  {
	  "if_name": "*<if_name>*",
	  "vr_id": "*<vr_id>*",
	  "ip_addr": "*<ip_addr>*",
	  "ad_intvl": "*<ad_intvl>*",
	  "preempt": "*<preempt>*",
	  "prio": "*<prio>*",
	  "admin_state": "*<admin_state>*",
	  "oper_state": "*<oper_state>*",
	  "track_if": "*<track_if>*",
	  "accept_mode": "*<accept_mode>*",
	  "switch_back_delay": "*<switch_back_delay>*",
	  "v2_compt": "*<v2_compt>*"
	  }
	]

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`vr_id`	The VRRP session Virtual Router (VR) ID; an integer from *1-255*. Default value is *0*.
`ip_addr`	The IP address of the VR; a valid IPv4 address.
`ad_intvl`	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from *5-4095*. Default value: *100* centi-seconds.
`preempt`	Enable the preemption of a lower priority master; one of *yes, no*. Default value: *yes*.
`prio`	The priority of the VR on the switch; an integer from *1-254*. Default value: *100*.
`admin_state`	Enable the VR one of *up, down*. Default value: *up*.

Element	Description
`oper_state`	The operation state of the VR; one of *master, backup, init*.
`track_if`	The interface to track by this VR. Default value: *none*. **Note:** If an interface is specified, it must exist.
`accept_mode`	Enables or disables the accept mode for this session; one of *yes, no*. Default value: *yes*.
`switch_back_delay`	The switch back delay interval; an integer from *1-500000*, or *0* to disable (default).
`v2_compt`	Enables backward compatibility for VRRPv2 for the VR; one of *yes, no*. Default value: *no*.

Get VRRP VRs of One Interface

Gets properties of all VRRP VRs under one specified interface.

Request

Method Type	GET
Request URI	/nos/api/cfg/vrrp/<*if_name*>
Request Body (JSON)	

where:

Element	Description
if_name	Interface name. **Note:** The interface must exist.

Response

Response Body
 	
(JSON)
	[
	{
	"if_name": "<*if_name*>",
	"vr_id": "<*vr_id*>",
	"ip_addr": "<*ip_addr*>",
	"ad_intvl": "<*ad_intvl*>",
	"preempt": "<*preempt*>",
	"prio": "<*prio*>",
	"admin_state": "<*admin_state*>",
	"oper_state": "<*oper_state*>",
	"track_if": "<*track_if*>",
	"accept_mode": "<*accept_mode*>",
	"switch_back_delay": "<*switch_back_delay*>",
	"v2_compt": "<*v2_compt*>"
	}
	]

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`vr_id`	Virtual Router (VR) identifier; an integer from *1-255*.
`ip_addr`	The IP address of the VR; a valid IPv4 address.
`ad_intvl`	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from *5-4095*. Default value: *100* centi-seconds.

Element	Description
`preempt`	Enable the preemption of a lower priority master; one of *yes, no*. Default value: *yes*.
`prio`	The priority of the VR on the switch; an integer from *1-254*. Default value: *100*.
`admin_state`	Enable the VR one of *up, down*. Default value: *up*.
`oper_state`	The operation state of the VR; one of *master, backup, init*.
`track_if`	The interface to track by this VR. Default value: *none*. **Note:** If an interface is specified, it must exist.
`accept_mode`	Enables or disables the accept mode for this session; one of *yes, no*. Default value: *yes*.
`switch_back_delay`	The switch back delay interval; an integer from *1-500000*, or *0* to disable (default).
`v2_compt`	Enables backward compatibility for VRRPv2 for the VR; one of *yes, no*. Default value: *no*.

Create VRRP VR

Creates a VRRP VR.

Request

Method Type	POST
Request URI	/nos/api/cfg/vrrp/<*if_name*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"vr_id": "<*vr_id*>",
	"ip_addr": "<*ip_addr*>",
	"ad_intvl": "<*ad_intvl*>",
	"preempt": "<*preempt*>",
	"prio": "<*prio*>",
	"admin_state": "<*admin_state*>",
	"track_if": "<*track_if*>",
	"accept_mode": "<*accept_mode*>",
	"switch_back_delay": "<*switch_back_delay*>",
	"v2_compt": "<*v2_compt*>"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`vr_id`	Virtual Router (VR) identifier; an integer from *1-255*.
`ip_addr`	The IP address of the VR; a valid IPv4 address.
`ad_intvl`	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from *5-4095*. Default value: *100* centi-seconds.
`preempt`	Enable the preemption of a lower priority master; one of *yes, no*. Default value: *yes*.
`prio`	The priority of the VR on the switch; an integer from *1-254*. Default value: *100*.
`admin_state`	Enable the VR one of *up, down*. Default value: *up*.
`oper_state`	The operation state of the VR; one of *master, backup, init*.
`track_if`	The interface to track by this VR. Default value: ■□■ℕ. **Note:** If an interface is specified, it must exist.
`accept_mode`	Enables or disables the accept mode for this session; one of *yes, no*. Default value: *yes*.

Element	Description
switch_back_delay	The switch back delay interval; an integer from *1-500000,* or *0* to disable (default).
v2_compt	Enables backward compatibility for VRRPv2 for the VR; one of *yes, no*. Default value: *no*.

Response

Response Body
 	
(JSON)
	[
	  {
	  "if_name": "*<if_name>*",
	  "vr_id": "*<vr_id>*",
	  "ip_addr": "*<ip_addr>*",
	  "ad_intvl": "*<ad_intvl>*",
	  "preempt": "*<preempt>*",
	  "prio": "*<prio>*",
	  "admin_state": "*<admin_state>*",
	  "oper_state": "*<oper_state>*",
	  "track_if": "*<track_if>*",
	  "accept_mode": "*<accept_mode>*",
	  "switch_back_delay": "*<switch_back_delay>*",
	  "v2_compt": "*<v2_compt>*"
	  }
	]

where:

Element	Description
if_name	Interface name. **Note:** The interface must exist.
vr_id	Virtual Router (VR) identifier; an integer from *1-255*.
ip_addr	The IP address of the VR; a valid IPv4 address.
ad_intvl	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from *5-4095*. Default value: *100* centi-seconds.
preempt	Enable the preemption of a lower priority master; one of *yes, no*. Default value: *yes*.
prio	The priority of the VR on the switch; an integer from *1-254*. Default value: *100*.
admin_state	Enable the VR one of *up, down.* Default value: *up*.
oper_state	The operation state of the VR; one of *master, backup, init*.
track_if	The interface to track by this VR. Default value: *none*. **Note:** If an interface is specified, it must exist.
accept_mode	Enables or disables the accept mode for this session; one of *yes, no*. Default value: *yes*.

Element	Description
`switch_back_delay`	The switch back delay interval; an integer from *1-500000,* or *0* to disable (default).
`v2_compt`	Enables backward compatibility for VRRPv2 for the VR; one of *yes, no*. Default value: *no*.

Get VRRP VR

Gets properties of a VRRP VR.

Request

Method Type	GET
Request URI	/nos/api/cfg/vrrp/<*if_name*>/<*vrid*>
Request Body (JSON)	

Response

Response Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"vr_id": "<*vr_id*>",
	"ip_addr": "<*ip_addr*>",
	"ad_intvl": "<*ad_intvl*>",
	"preempt": "<*preempt*>",
	"prio": "<*prio*>",
	"admin_state": "<*admin_state*>",
	"oper_state": "<*oper_state*>",
	"track_if": "<*track_if*>",
	"accept_mode": "<*accept_mode*>",
	"switch_back_delay": "<*switch_back_delay*>",
	"v2_compt": "<*v2_compt*>"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`vr_id`	Virtual Router (VR) identifier; an integer from *1-255.*
`ip_addr`	The IP address of the VR; a valid IPv4 address.
`ad_intvl`	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from *5-4095*. Default value: *100* centi-seconds.
`preempt`	Enable the preemption of a lower priority master; one of *yes, no*. Default value: *yes*.
`prio`	The priority of the VR on the switch; an integer from *1-254*. Default value: 100.
`admin_state`	Enable the VR one of *up, down*. Default value: *up*.
`oper_state`	The operation state of the VR; one of *master, backup, init*.

Element	Description
`track_if`	The interface to track by this VR. Default value: *none*. **Note:** If an interface is specified, it must exist.
`accept_mode`	Enables or disables the accept mode for this session; one of *yes, no*. Default value: *yes*.
`switch_back_delay`	The switch back delay interval; an integer from *1-500000,* or *0* to disable (default).
`v2_compt`	Enables backward compatibility for VRRPv2 for the VR; one of *yes, no*. Default value: *no*.

Update VRRP VR

Updates the properties of a VRRP VR.

Request

Method Type	PUT
Request URI	/nos/api/cfg/vrrp/<*if_name*>/<*vrid*>

Request Body
 	
(JSON)
	{
	"if_name": "<*if_name*>",
	"vr_id": "<*vr_id*>",
	"ip_addr": "<*ip_addr*>",
	"ad_intvl": "<*ad_intvl*>",
	"preempt": "<*preempt*>",
	"prio": "<*prio*>",
	"admin_state": "<*admin_state*>",
	"track_if": "<*track_if*>",
	"accept_mode": "<*accept_mode*>",
	"switch_back_delay": "<*switch_back_delay*>",
	"v2_compt": "<*v2_compt*>"
	}

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`vr_id`	Virtual Router (VR) identifier; an integer from *1-255*.
`ip_addr`	The IP address of the VR; a valid IPv4 address.
`ad_intvl`	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from *5-4095*. Default value: *100* centi-seconds.
`preempt`	Enable the preemption of a lower priority master; one of *yes, no*. Default value: *yes*.
`prio`	The priority of the VR on the switch; an integer from *1-254*. Default value: *100*.
`admin_state`	Enable the VR; one of *up, down*. Default value: *up*.
`oper_state`	The operation state of the VR; one of *master, backup, init*.
`track_if`	The interface to track by this VR. Default value: *none*. **Note:** If an interface is specified, it must exist.
`accept_mode`	Enables or disables the accept mode for this session; one of *yes, no*. Default value: *yes*.

Element	Description
`switch_back_delay`	The switch back delay interval; an integer from *1-500000*, or *0* to disable (default).
`v2_compt`	Enables backward compatibility for VRRPv2 for the VR; one of *yes, no*. Default value: *no*.

Response

Response Body

(JSON)

```
{
    "if_name": "<if_name>",
    "vr_id": "<vr_id>",
    "ip_addr": "<ip_addr>",
    "ad_intvl": "<ad_intvl>",
    "preempt": "<preempt>",
    "prio": "<prio>",
    "admin_state": "<admin_state>",
    "oper_state": "<oper_state>",
    "track_if": "<track_if>",
    "accept_mode": "<accept_mode>",
    "switch_back_delay": "<switch_back_delay>",
    "v2_compt": "<v2_compt>"
}
```

where:

Element	Description
`if_name`	Interface name. **Note:** The interface must exist.
`vr_id`	Virtual Router (VR) identifier; an integer from *1-255*.
`ip_addr`	The IP address of the VR; a valid IPv4 address.
`ad_intvl`	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from *5-4095*. Default value: *100* centi-seconds.
`preempt`	Enable the preemption of a lower priority master; one of *yes, no*. Default value: *yes*.
`prio`	The priority of the VR on the switch; an integer from *1-254*. Default value: *100*.
`admin_state`	Enable the VR; one of *up, down*. Default value: *up*.
`oper_state`	The operation state of the VR; one of *master, backup, init.*
`track_if`	The interface to track by this VR. Default value: *none*. **Note:** If an interface is specified, it must exist.
`accept_mode`	Enables or disables the accept mode for this session; one of *yes, no*. Default value: *yes*.

Element	Description
`switch_back_delay`	The switch back delay interval; an integer from *1-500000*, or *0* to disable (default).
`v2_compt`	Enables backward compatibility for VRRPv2 for the VR; one of *yes, no*. Default value: *no*.

Delete VRRP VR

Deletes a VRRP VR.

Note: If the specified *vrid* is `all`, all VRRP VRs entries in the specified interface will be deleted.

Request

Method Type	DELETE
Request URI	/nos/api/cfg/vrrp/<*if_name*>/<*vrid*>
Request Body (JSON)	

Appendix A. Getting Help and Technical Assistance

If you need help, service, or technical assistance or just want more information about Lenovo products, you will find a wide variety of sources available from Lenovo to assist you.

Use this information to obtain additional information about Lenovo and Lenovo products, and determine what to do if you experience a problem with your Lenovo system or optional device.

Note: This section includes references to IBM web sites and information about obtaining service. IBM is Lenovo's preferred service provider for the System x, Flex System, and NeXtScale System products.

Before you call, make sure that you have taken these steps to try to solve the problem yourself.

If you believe that you require warranty service for your Lenovo product, the service technicians will be able to assist you more efficiently if you prepare before you call.

- Check all cables to make sure that they are connected.
- Check the power switches to make sure that the system and any optional devices are turned on.
- Check for updated software, firmware, and operating-system device drivers for your Lenovo product. The Lenovo Warranty terms and conditions state that you, the owner of the Lenovo product, are responsible for maintaining and updating all software and firmware for the product (unless it is covered by an additional maintenance contract). Your service technician will request that you upgrade your software and firmware if the problem has a documented solution within a software upgrade.
- If you have installed new hardware or software in your environment, check the Lenovo ServerProven website to make sure that the hardware and software is supported by your product.
- Go to the Lenovo Support portal to check for information to help you solve the problem.
- Gather the following information to provide to the service technician. This data will help the service technician quickly provide a solution to your problem and ensure that you receive the level of service for which you might have contracted.
  - Hardware and Software Maintenance agreement contract numbers, if applicable
  - Machine type number (if applicable–Lenovo 4-digit machine identifier)
  - Model number
  - Serial number
  - Current system UEFI and firmware levels
  - Other pertinent information such as error messages and logs

- Start the process of determining a solution to your problem by making the pertinent information available to the service technicians. The IBM service technicians can start working on your solution as soon as you have completed and submitted an Electronic Service Request.

You can solve many problems without outside assistance by following the troubleshooting procedures that Lenovo provides in the online help or in the Lenovo product documentation. The Lenovo product documentation also describes the diagnostic tests that you can perform. The documentation for most systems, operating systems, and programs contains troubleshooting procedures and explanations of error messages and error codes. If you suspect a software problem, see the documentation for the operating system or program.

Appendix B. Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area.

Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
1009 Think Place - Building One
Morrisville, NC 27560
U.S.A.

Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties.

Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Trademarks

Lenovo, the Lenovo logo, Flex System, System x, NeXtScale System, and X-Architecture are trademarks of Lenovo in the United States, other countries, or both.

Intel and Intel Xeon are trademarks of Intel Corporation in the United States, other countries, or both.

Internet Explorer, Microsoft, and Windows are trademarks of the Microsoft group of companies.

Linux is a registered trademark of Linus Torvalds.

Other company, product, or service names may be trademarks or service marks of others.

Important Notes

Processor speed indicates the internal clock speed of the microprocessor; other factors also affect application performance.

CD or DVD drive speed is the variable read rate. Actual speeds vary and are often less than the possible maximum.

When referring to processor storage, real and virtual storage, or channel volume, KB stands for 1 024 bytes, MB stands for 1 048 576 bytes, and GB stands for 1 073 741 824 bytes.

When referring to hard disk drive capacity or communications volume, MB stands for 1 000 000 bytes, and GB stands for 1 000 000 000 bytes. Total user-accessible capacity can vary depending on operating environments.

Maximum internal hard disk drive capacities assume the replacement of any standard hard disk drives and population of all hard-disk-drive bays with the largest currently supported drives that are available from Lenovo.

Maximum memory might require replacement of the standard memory with an optional memory module.

Each solid-state memory cell has an intrinsic, finite number of write cycles that the cell can incur. Therefore, a solid-state device has a maximum number of write cycles that it can be subjected to, expressed as total bytes written (TBW). A device that has exceeded this limit might fail to respond to system-generated commands or might be incapable of being written to. Lenovo is not responsible for replacement of a device that has exceeded its maximum guaranteed number of program/erase cycles, as documented in the Official Published Specifications for the device.

Lenovo makes no representations or warranties with respect to non-Lenovo products. Support (if any) for the non-Lenovo products is provided by the third party, not Lenovo.

Some software might differ from its retail version (if available) and might not include user manuals or all program functionality.

Recycling Information

Lenovo encourages owners of information technology (IT) equipment to responsibly recycle their equipment when it is no longer needed. Lenovo offers a variety of programs and services to assist equipment owners in recycling their IT products. For information on recycling Lenovo products, go to:

http://www.lenovo.com/recycling

Particulate Contamination

Attention: Airborne particulates (including metal flakes or particles) and reactive gases acting alone or in combination with other environmental factors such as humidity or temperature might pose a risk to the device that is described in this document.

Risks that are posed by the presence of excessive particulate levels or concentrations of harmful gases include damage that might cause the device to malfunction or cease functioning altogether. This specification sets forth limits for particulates and gases that are intended to avoid such damage. The limits must not be viewed or used as definitive limits, because numerous other factors, such as temperature or moisture content of the air, can influence the impact of particulates or environmental corrosives and gaseous contaminant transfer. In the absence of specific limits that are set forth in this document, you must implement practices that maintain particulate and gas levels that are consistent with the protection of human health and safety. If Lenovo determines that the levels of particulates or gases in your environment have caused damage to the device, Lenovo may condition provision of repair or replacement of devices or parts on implementation of appropriate remedial measures to mitigate such environmental contamination. Implementation of such remedial measures is a customer responsibility..

Contaminant	Limits
Particulate	• The room air must be continuously filtered with 40% atmospheric dust spot efficiency (MERV 9) according to ASHRAE Standard 52.2[1]. • Air that enters a data center must be filtered to 99.97% efficiency or greater, using high-efficiency particulate air (HEPA) filters that meet MIL-STD-282. • The deliquescent relative humidity of the particulate contamination must be more than 60%[2]. • The room must be free of conductive contamination such as zinc whiskers.
Gaseous	• Copper: Class G1 as per ANSI/ISA 71.04-1985[3] • Silver: Corrosion rate of less than 300 Å in 30 days

[1] ASHRAE 52.2-2008 - *Method of Testing General Ventilation Air-Cleaning Devices for Removal Efficiency by Particle Size*. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

[2] The deliquescent relative humidity of particulate contamination is the relative humidity at which the dust absorbs enough water to become wet and promote ionic conduction.

[3] ANSI/ISA-71.04-1985. *Environmental conditions for process measurement and control systems: Airborne contaminants*. Instrument Society of America, Research Triangle Park, North Carolina, U.S.A.

Telecommunication Regulatory Statement

This product may not be certified in your country for connection by any means whatsoever to interfaces of public telecommunications networks. Further certification may be required by law prior to making any such connection. Contact a Lenovo representative or reseller for any questions.

Electronic Emission Notices

When you attach a monitor to the equipment, you must use the designated monitor cable and any interference suppression devices that are supplied with the monitor.

Federal Communications Commission (FCC) Statement

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Properly shielded and grounded cables and connectors must be used to meet FCC emission limits. Lenovo is not responsible for any radio or television interference caused by using other than recommended cables and connectors or by unauthorized changes or modifications to this equipment. Unauthorized changes or modifications could void the user's authority to operate the equipment.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that might cause undesired operation.

Industry Canada Class A Emission Compliance Statement

This Class A digital apparatus complies with Canadian ICES-003.

Avis de Conformité à la Réglementation d'Industrie Canada

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

Australia and New Zealand Class A Statement

Attention: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

European Union - Compliance to the Electromagnetic Compatibility Directive

This product is in conformity with the protection requirements of EU Council Directive 2004/108/EC (until April 19, 2016) and EU Council Directive 2014/30/EU (from April 20, 2016) on the approximation of the laws of the Member States relating to electromagnetic compatibility. Lenovo cannot accept responsibility for any failure to satisfy the protection requirements resulting from a non-recommended modification of the product, including the installation of option cards from other manufacturers.

This product has been tested and found to comply with the limits for Class A equipment according to European Standards harmonized in the Directives in compliance. The limits for Class A equipment were derived for commercial and industrial environments to provide reasonable protection against interference with licensed communication equipment.

CE Lenovo, Einsteinova 21, 851 01 Bratislava, Slovakia

Warning: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Germany Class A Statement

Deutschsprachiger EU Hinweis:

Hinweis für Geräte der Klasse A EU-Richtlinie zur Elektromagnetischen Verträglichkeit

Dieses Produkt entspricht den Schutzanforderungen der EU-Richtlinie 2014/30/EU (früher 2004/108/EC) zur Angleichung der Rechtsvorschriften über die elektromagnetische Verträglichkeit in den EU-Mitgliedsstaaten und hält die Grenzwerte der Klasse A der Norm gemäß Richtlinie.

Um dieses sicherzustellen, sind die Geräte wie in den Handbüchern beschrieben zu installieren und zu betreiben. Des Weiteren dürfen auch nur von der Lenovo empfohlene Kabel angeschlossen werden. Lenovo übernimmt keine Verantwortung für die Einhaltung der Schutzanforderungen, wenn das Produkt ohne Zustimmung der Lenovo verändert bzw. wenn Erweiterungskomponenten von Fremdherstellern ohne Empfehlung der Lenovo gesteckt/eingebaut werden.

Deutschland:

Einhaltung des Gesetzes über die elektromagnetische Verträglichkeit von Betriebsmittein

Dieses Produkt entspricht dem „Gesetz über die elektromagnetische Verträglichkeit von Betriebsmitteln“ EMVG (früher „Gesetz über die elektromagnetische Verträglichkeit von Geräten“). Dies ist die Umsetzung der EU-Richtlinie 2014/30/EU (früher 2004/108/EC) in der Bundesrepublik Deutschland.

Zulassungsbescheinigung laut dem Deutschen Gesetz über die elektromagnetische Verträglichkeit von Betriebsmitteln, EMVG vom 20. Juli 2007 (früher Gesetz über die elektromagnetische Verträglichkeit von Geräten), bzw. der EMV EU Richtlinie 2014/30/EU (früher 2004/108/EC), für Geräte der Klasse A.

Dieses Gerät ist berechtigt, in Übereinstimmung mit dem Deutschen EMVG das EG-Konformitätszeichen - CE - zu führen. Verantwortlich für die Konformitätserklärung nach Paragraf 5 des EMVG ist die Lenovo (Deutschland) GmbH, Meitnerstr. 9, D-70563 Stuttgart.

Informationen in Hinsicht EMVG Paragraf 4 Abs. (1) 4:

Das Gerät erfüllt die Schutzanforderungen nach EN 55024 und EN 55022 Klasse A.

Nach der EN 55022: „Dies ist eine Einrichtung der Klasse A. Diese Einrichtung kann im Wohnbereich Funkstörungen verursachen; in diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen durchzuführen und dafür aufzukommen."

Nach dem EMVG: „Geräte dürfen an Orten, für die sie nicht ausreichend entstört sind, nur mit besonderer Genehmigung des Bundesministers für Post und Telekommunikation oder des Bundesamtes für Post und Telekommunikation betrieben werden. Die Genehmigung wird erteilt, wenn keine elektromagnetischen Störungen zu erwarten sind." (Auszug aus dem EMVG, Paragraph 3, Abs. 4). Dieses Genehmigungsverfahrenist nach Paragraph 9 EMVG in Verbindung mit der entsprechenden Kostenverordnung (Amtsblatt 14/93) kostenpflichtig.

Anmerkung: Um die Einhaltung des EMVG sicherzustellen sind die Geräte, wie in den Handbüchern angegeben, zu installieren und zu betreiben.

Japan VCCI Class A Statement

この装置は、クラス A 情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。 VCCI–A

This is a Class A product based on the standard of the Voluntary Control Council for Interference (VCCI). If this equipment is used in a domestic environment, radio interference may occur, in which case the user may be required to take corrective actions.

Japan Electronics and Information Technology Industries Association (JEITA) Statement

高調波ガイドライン適合品

Japan Electronics and Information Technology Industries Association (JEITA) Confirmed Harmonics Guidelines (products less than or equal to 20 A per phase)

高調波ガイドライン準用品

Japan Electronics and Information Technology Industries Association (JEITA) Confirmed Harmonics Guidelines with Modifications (products greater than 20 A per phase).

Korea Communications Commission (KCC) Statement

이 기기는 업무용(A급)으로 전자파적합기기로
서 판매자 또는 사용자는 이 점을 주의하시기
바라며, 가정외의 지역에서 사용하는 것을 목
적으로 합니다.

This is electromagnetic wave compatibility equipment for business (Type A). Sellers and users need to pay attention to it. This is for any areas other than home.

Russia Electromagnetic Interference (EMI) Class A statement

ВНИМАНИЕ! Настоящее изделие относится к классу А.
В жилых помещениях оно может создавать радиопомехи, для снижения которых необходимы дополнительные меры

People's Republic of China Class A electronic emission statement

中华人民共和国“A类”警告声明

声 明

此为A级产品，在生活环境中，该产品可能会造成无线电干扰。在这种情况下，可能需要用户对其干扰采取切实可行的措施。

Taiwan Class A compliance statement

警告使用者:
這是甲類的資訊產品，在居住的環境中使用時，可能會造成射頻干擾，在這種情況下，使用者會被要求採取某些適當的對策。