

Lenovo Network

# REST API Programming Guide

For Lenovo Cloud Network Operating System 10.3

**Lenovo**<sup>TM</sup>

**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.

Third Edition (February 2017)

© Copyright Lenovo 2017  
Portions © Copyright IBM Corporation 2014.

LIMITED AND RESTRICTED RIGHTS NOTICE: If data or software is delivered pursuant a General Services Administration "GSA" contract, use, reproduction, or disclosure is subject to restrictions set forth in Contract No. GS-35F-05925.

Lenovo and the Lenovo logo are trademarks of Lenovo in the United States, other countries, or both.

---

# Contents

<b>Preface</b> . . . . .	<b>.9</b>
Who Should Use This Guide . . . . .	.10
What You'll Find in This Guide. . . . .	.11
Additional References . . . . .	.12
Typographic Conventions . . . . .	.13
<b>Chapter 1. Introduction</b> . . . . .	<b>15</b>
REST API Components . . . . .	.16
Using the REST Server. . . . .	.17
Starting and Stopping the REST Server. . . . .	.17
Communicating with the REST Server . . . . .	.17
Server Security . . . . .	.19
HTTPS Support . . . . .	.20
REST Server Limitations . . . . .	.22
<b>Chapter 2. REST Server JSON Calls</b> . . . . .	<b>23</b>
System . . . . .	.24
Get System Properties . . . . .	.24
System Information . . . . .	.25
Get All System Information. . . . .	.26
Get System Fan Information . . . . .	.29
Get System Power Information . . . . .	.30
Get System Temperature Information . . . . .	.31
Get System Inventory . . . . .	.32
Get System Serial Number . . . . .	.33
Get Panic Dump Information . . . . .	.34
Startup Information . . . . .	.35
Get System Startup Information. . . . .	.36
Put System ZTP Interface . . . . .	.37
Get System ZTP Setting . . . . .	.38
Put System Startup Image . . . . .	.39
Get System Startup Image . . . . .	.40
NOS Copy . . . . .	.41
Determine Whether the Running Configuration Needs to be Saved . . . . .	.42
Reset Switch . . . . .	.43
Save Configuration . . . . .	.44
Download Image to Switch. . . . .	.45
Download Configuration to Switch . . . . .	.46
Upload Configuration to Server. . . . .	.48
Upload Tech Support to Server . . . . .	.50
Get Download Transfer Status . . . . .	.51
Get Upload Transfer Status. . . . .	.52

System Configuration . . . . .	54
Get Hostname . . . . .	55
Set Hostname . . . . .	56
Get Clock Date . . . . .	57
Set Clock Date . . . . .	58
Set Clock Format . . . . .	60
Set Clock Protocol. . . . .	61
Set Clock Timezone . . . . .	62
Set Clock Summer Time . . . . .	63
Get Device Contact . . . . .	66
Update Device Contact . . . . .	67
Get Device Description . . . . .	68
Update Device Description. . . . .	69
Get Rack Properties . . . . .	70
Update Rack Properties . . . . .	71
Interface. . . . .	72
Get All Interfaces . . . . .	73
Get Interface . . . . .	74
Update Interface . . . . .	75
Get Transceiver Information for All Interfaces . . . . .	77
Get Transceiver Information for One Interfaces . . . . .	79
LACP. . . . .	81
Get LACP System Properties . . . . .	82
Update LACP System Properties . . . . .	83
LAG . . . . .	84
Get All LAGs. . . . .	85
Create LAG . . . . .	87
Get LAG Properties . . . . .	89
Update LAG . . . . .	91
Get LAG Load Balance Settings. . . . .	93
Update LAG Load Balance Settings . . . . .	94
Delete LAG . . . . .	96
vLAG. . . . .	97
Get vLAG Configuration. . . . .	98
Update vLAG Configuration . . . . .	99
Get Global vLAG Information . . . . .	101
Get vLAG ISL Information . . . . .	103
Configure vLAG ISL . . . . .	104
Get vLAG Health Check Information . . . . .	105
Configure vLAG Health Check Parameters. . . . .	106
Create vLAG Instance . . . . .	108
Update vLAG Instance . . . . .	109
Delete vLAG Instance . . . . .	110
Get vLAG Instance Configuration. . . . .	111
Get vLAG Instance Information. . . . .	112

VLAN. . . . .	113
Get All VLANs . . . . .	114
Create VLAN. . . . .	115
Get VLAN . . . . .	116
Update VLAN . . . . .	117
Delete VLAN. . . . .	118
VLAN Interface Properties . . . . .	119
Get VLAN Properties of All Interfaces . . . . .	120
Get VLAN Interface Properties . . . . .	121
Update VLAN Interface Properties . . . . .	122
STP . . . . .	123
Get STP Properties for All Interfaces . . . . .	124
Get STP Interface Properties . . . . .	125
Update STP Interface Properties. . . . .	126
MSTP . . . . .	128
Get MSTP System Properties . . . . .	129
Update MSTP System Properties . . . . .	130
Get Properties of All MSTP Instances . . . . .	131
Create MSTP Instance . . . . .	132
Get MSTP Instance . . . . .	133
Update MSTP Instance . . . . .	134
Delete MSTP Instance . . . . .	135
Get Interface Properties of an MSTP Instance . . . . .	136
Update Interface Properties of an MSTP Instance . . . . .	137
LLDP . . . . .	138
Get LLDP System Properties . . . . .	139
Update LLDP System Properties . . . . .	140
Get LLDP Properties for All Interfaces . . . . .	141
Get LLDP Interface Properties . . . . .	142
Update LLDP Interface Properties. . . . .	143
Get LLDP Interface Statistics . . . . .	144
Get LLDP Interface Neighbor information . . . . .	145
Get LLDP Neighbor Information for All Interfaces. . . . .	146
VRF. . . . .	147
Get All VRFs . . . . .	148
Get VRF . . . . .	149
IP Interface . . . . .	150
Get IP Properties of All Interfaces . . . . .	151
Get IP Interface Properties . . . . .	152
Update IP Interface Properties . . . . .	153
ARP . . . . .	154
Get ARP System Properties . . . . .	155
Update ARP System Properties . . . . .	156
Get ARP Properties of All Interfaces . . . . .	157
Get ARP Interface Properties . . . . .	158
Update ARP Interface Properties . . . . .	159

Static ARP . . . . .	160
Get Static ARP Entries of All Interfaces . . . . .	161
Get Static ARP Entries of One Interface . . . . .	162
Create Static ARP Entry . . . . .	163
Get Static ARP Entry . . . . .	164
Update Static ARP Entry . . . . .	165
Delete Static ARP Entry . . . . .	166
VRRP . . . . .	167
Get VRRP VRs of All Interfaces . . . . .	168
Get VRRP VRs of One Interface . . . . .	170
Create VRRP VR . . . . .	172
Get VRRP VR . . . . .	175
Update VRRP VR . . . . .	177
Delete VRRP VR . . . . .	180
IGMP . . . . .	181
Get IGMP Groups . . . . .	182
Get IGMP Mrouter . . . . .	185
Update IGMP Mrouter Interface for a VLAN . . . . .	186
Get IGMP Querier . . . . .	187
Update IGMP Querier on a VLAN . . . . .	188
Get IGMP Snooping System Properties . . . . .	189
Update IGMP Snooping System Properties . . . . .	190
Get IGMP Snooping Properties of All VLANs . . . . .	191
Get IGMP Snooping VLAN Properties . . . . .	192
Update IGMP Snooping VLAN Properties . . . . .	193
SNMP . . . . .	195
Get the SNMPv3 Account for XClarity . . . . .	196
Set the SNMPv3 Account for XClarity . . . . .	197
Delete the SNMPv3 Account for XClarity . . . . .	199
Get the SNMPv3 Trap Host IP Address for XClarity . . . . .	200
Set the SNMPv3 Trap Host IP Address for XClarity . . . . .	201
Delete the SNMPv3 Trap Host IP Address for XClarity . . . . .	203

BGP. . . . .	204
Get BGP Global Statistics. . . . .	206
Clear BGP Global Statistics . . . . .	208
Get BGP Neighbor Received RIB Information . . . . .	209
Get BGP Neighbor RIB Advertised Information . . . . .	211
Get BGP Global Configuration . . . . .	213
Get BGP Best Path Configuration . . . . .	215
Get BGP Confederation Configuration . . . . .	217
Get BGP Graceful-Restart Configuration . . . . .	218
Get BGP Route Reflector Information . . . . .	219
Get BGP RIB Information . . . . .	220
Get BGP Detailed RIB Information . . . . .	222
Get BGP Summary Information . . . . .	226
Get BGP Neighbor Details . . . . .	228
Get BGP Neighbor Statistics . . . . .	233
Get BGP Distance Configuration . . . . .	234
Get BGP Address Family Global Configuration . . . . .	235
Get BGP Multipath ECMP Numbers Configuration . . . . .	236
Get BGP Nexthop Trigger-Delay Configuration . . . . .	237
Get BGP Aggregate Configuration. . . . .	238
Get BGP Dampening Parameters Configuration. . . . .	239
Get BGP Dampened Path Configuration . . . . .	241
Get BGP Network Configuration . . . . .	243
Get BGP Redistribute Configuration . . . . .	244
Get BGP Neighbor Configuration . . . . .	246
DHCP. . . . .	251
Get the Global DHCP Client Feature Property . . . . .	252
Update DHCP Client Feature Property. . . . .	253
Get DHCP Client Properties of All Interfaces . . . . .	254
Update DHCP Client Interface Properties . . . . .	255
Delete the Vendor Class Identifier of an Interface . . . . .	257
Get the Global DHCP Relay Service Property . . . . .	258
Update the Global DHCP Relay Service Property . . . . .	259
Get the DHCP Relay Properties of All Interfaces. . . . .	260
Get DHCP Relay Interface Properties . . . . .	261
Update DHCP Relay Interface Properties. . . . .	263
Delete DHCP Relay Interface Properties . . . . .	265
STP. . . . .	267
Get STP Properties Per VLAN . . . . .	268
Set STP Properties Per VLAN. . . . .	269
Get STP Interface Properties . . . . .	270
Update STP Interface Properties . . . . .	271
Get STP Interface VLAN Properties . . . . .	272
Update STP Interface VLAN Properties . . . . .	273
<b>Appendix A. Getting Help and Technical Assistance . . . . .</b>	<b>275</b>
<b>Appendix B. Notices. . . . .</b>	<b>277</b>
Trademarks . . . . .	279
Important Notes . . . . .	280

Recycling Information. . . . .	281
Particulate Contamination. . . . .	282
Telecommunication Regulatory Statement . . . . .	283
Electronic Emission Notices . . . . .	284
Federal Communications Commission (FCC) Statement . . . . .	284
Industry Canada Class A Emission Compliance Statement . . . . .	284
Avis de Conformité à la Réglementation d'Industrie Canada . . . . .	284
Australia and New Zealand Class A Statement . . . . .	284
European Union EMC Directive Conformance Statement. . . . .	284
Germany Class A Statement . . . . .	285
Japan VCCI Class A Statement . . . . .	286
Japan Electronics and Information Technology Industries Association (JEITA) Statement. . . . .	287
Korea Communications Commission (KCC) Statement. . . . .	287
Russia Electromagnetic Interference (EMI) Class A statement . . . . .	288
People's Republic of China Class A electronic emission statement. . . . .	289
Taiwan Class A compliance statement. . . . .	290



---

# Preface

The *REST API Programming Guide for Lenovo CNOS 10.2* describes how to configure and use the Cloud Network Operating System 10.3 software on your Lenovo switch. For documentation on installing the switch physically, see the *Installation Guide* for your switch.

---

## 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 switch Installation Guide*
- *Lenovo Network Application Guide for Lenovo Cloud Network Operating System 10.3*
- *Lenovo Network Command Reference for Lenovo Cloud Network Operating System 10.3*

# 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 <code>readme.txt</code> file.  Main#
<b>ABC123</b>	This bold type appears in command examples. It shows text that must be typed in exactly as shown.	Main# <b>sys</b>
<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: host# <b>telnet</b> <IP address>  Read your <i>User's Guide</i> thoroughly.
[ ]	Command items shown inside brackets are optional and can be used or excluded as the situation demands. Do not type the brackets.	host# <b>ls</b> [-a]
	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.	host# <b>set left right</b>
<b>AaBbCc123</b>	This block type depicts menus, buttons, and other controls that appear in Web browsers and other graphical interfaces.	Click the <b>Save</b> button.



---

# Chapter 1. Introduction

The Lenovo REST Application Programming Interface (API) enables you to remotely configure and manage a Lenovo switch using REST, and 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.

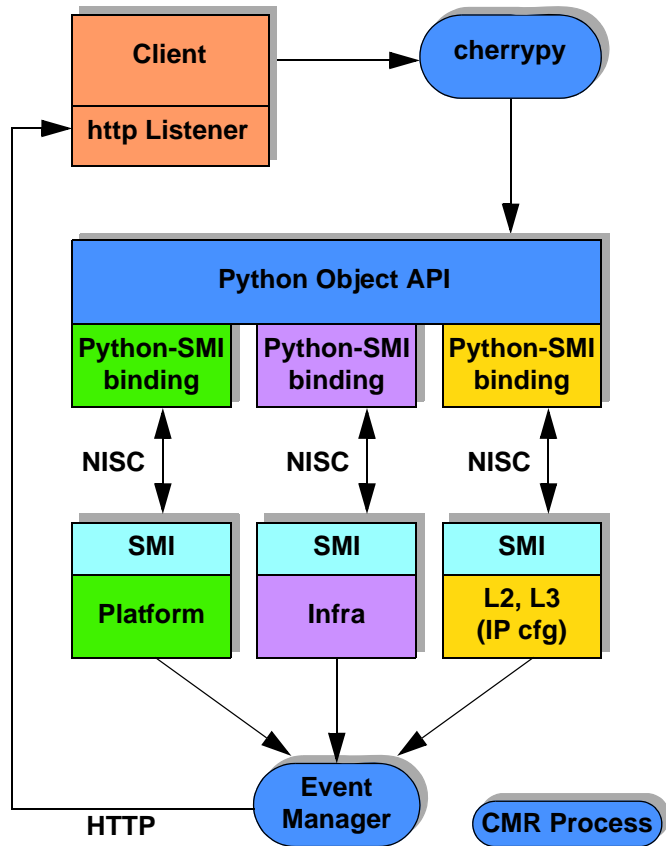
**Note:** The Lenovo REST API calls have been tested with:

- 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 Command Mode on the switch, enter

```
SWITCH(config)# feature restApi
```

To start the REST server listening in HTTP mode on port 8090, in Command 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 Command 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
<i>switch address:port</i>	The switch IP address and port where the REST server is installed.
<i>resource</i>	Any network or switch resource, such as an interface or a VLAN.
<i>parameters</i>	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:

```
display restApi server
```

## Authenticating Users on the REST Server

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

```
https://<ipaddress>/nos/api/login
```

where *ipaddress* 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://<ipaddress>/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

To enable HTTPS over the REST server, you need to generate a certificate using 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. Create the certificate:

```
SWITCH(config)# access generate-certificate vrf management
WARNING :Disable REST server before generating a new certificate!!
Country Name (2 letter code) [US]:
State or Province Name (full name) [CA]:
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) []: Lynn Gold
Email (eg, email address) []: netuser@lenovo.com
Confirm generate certificate? (y/n) [n] y
```

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

3. Re-enable the REST server:

```
CNOS_Jupiter1_Auto(config)# feature restApi
```

4. Make sure the REST server is running:

```
CNOS_Jupiter1_Auto(config)# display restApi server
rest server enabled port: 443
```

5. Make sure the access host-certificate was generated:

```
CNOS_Jupiter1_Auto(config)# display access host-certificate vrf
management
Certificate:
  Data:
    Version: 3 (0x2)
    Serial Number:
      f4:66:2c:e4:b7:c9:56:eb
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: C=US, ST=CA, L=Santa Clara, O=Lenovo Networking Operating
System, OU=Network Engineering, CN=Lynn
Gold/emailAddress=netuser@lenovo.com
  Validity
    Not Before: Nov  8 01:30:54 2016 GMT
    Not After : Nov  8 01:30:54 2017 GMT
```

Subject: C=US, ST=CA, L=Santa Clara, O=Lenovo Networking  
Operating System, OU=Network Engineering, CN=Lynn  
Gold/emailAddress=netuser@lenovo.com

Subject Public Key Info:

Public Key Algorithm: rsaEncryption

Public-Key: (2048 bit)

Modulus:

00:b7:57:7d:21:c0:39:81:d3:73:0e:fe:f8:0e:de:  
5f:5d:4a:70:04:06:e3:d5:eb:f5:c5:8e:98:40:06:  
ab:21:79:43:26:4c:0b:21:6e:0f:62:42:09:d1:4c:  
6d:61:33:ae:a1:2e:a2:f2:b6:ac:07:cd:78:70:1b:  
df:41:aa:0a:0d:20:05:0f:0d:cc:e5:b0:9c:5b:be:  
c1:f6:fe:9f:ea:60:2e:7b:18:a8:64:65:36:68:c9:  
f9:e7:3d:21:e6:f7:c9:57:8e:36:26:19:b9:57:a3:  
e5:41:1d:d8:9b:42:5b:68:80:c1:db:01:67:ac:43:  
16:f4:56:d4:01:6f:22:04:08:12:7b:bb:2c:c5:8c:  
7d:8a:8a:06:42:99:c2:f8:45:bb:60:34:e5:11:c4:  
6d:d8:9f:47:2b:02:1c:6d:50:26:74:ba:7c:3c:97:  
5e:c8:d9:ba:03:e4:56:e3:c9:61:72:48:0e:17:ee:  
fe:e5:77:42:37:c1:8a:7e:75:08:16:ad:4b:fe:5e:  
10:80:b5:10:34:65:5d:79:1b:b4:34:83:15:06:45:  
b3:c6:f3:3a:e5:c2:1e:a7:7c:5f:b6:cf:a8:61:cf:  
87:65:d3:8a:17:f2:59:32:6d:db:5e:01:48:b1:b8:  
e3:ee:e5:14:e3:06:df:c9:39:f6:e7:42:aa:46:59:  
23:ef

Exponent: 65537 (0x10001)

X509v3 extensions:

X509v3 Subject Key Identifier:

FF:D0:EF:C0:4A:6D:DF:1C:F4:B7:B3:5E:DF:A5:94:0B:6B:3D:D9:AD

X509v3 Authority Key Identifier:

keyid:FF:D0:EF:C0:4A:6D:DF:1C:F4:B7:B3:5E:DF:A5:94:0B:6B:3D:D9:AD

X509v3 Basic Constraints:

CA:TRUE

Signature Algorithm: sha256WithRSAEncryption

84:05:92:2d:a8:80:13:55:d5:2d:86:91:b8:b9:1e:c6:22:37:  
95:d7:fb:ec:a7:26:99:cd:e0:09:15:54:98:01:ee:52:1a:44:  
f8:3f:a8:fd:28:dc:04:46:f6:36:74:5c:55:c3:ed:e9:84:1c:  
e7:21:1c:05:d7:5d:a7:79:bf:30:35:d3:41:a0:72:0a:39:3f:  
c1:c8:81:10:c8:e8:81:83:99:69:31:b6:9c:38:25:aa:07:47:  
12:15:d9:f6:b8:0c:44:a4:d2:da:5c:83:db:1a:ba:bb:74:ba:  
5a:f2:d8:7c:d7:37:35:14:04:d2:b3:24:3f:73:b5:0d:ce:4e:  
c2:f3:db:a2:57:d6:de:9d:c2:d0:ad:25:24:0d:8a:0c:83:fd:  
67:5f:5b:1e:f7:76:8c:cb:31:9a:5b:3b:be:a2:26:f9:ac:4b:  
45:0c:0c:c4:c2:91:c2:31:38:38:dd:54:6f:80:ce:3a:93:94:  
df:5f:75:de:7d:13:2d:ce:fb:ae:0f:1c:68:94:88:47:a1:5c:  
30:82:93:02:07:e6:cd:be:f9:e2:fb:10:a5:77:46:7a:75:5f:  
15:ca:81:6a:01:05:0d:c9:53:b8:cb:37:69:47:bd:84:e3:01:  
45:d1:8a:59:25:48:5f:ff:dd:6d:c2:7f:6a:b8:2f:96:d7:6b:  
b0:e9:21:78

---

## 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.

---

## System

The following system URI is available:

- /nos/api/system GET

### 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 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

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 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" : "30",
    "State" : "OK"
  },
  "Hot Spot" :
  {
    "Temp" : "46",
    "State": "OK"
  }
  "Temperature threshold" :
  {
    "System Warning" : 85,
    "System Shutdown" : 95,
    "System Set Point" : 70
  }
},
"Inventory" :
{
  "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"
},

```

	<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)	<pre>{   "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>
----------------------	--

where:

Element	Description
Module	Module number.
Air - flow	Air flow type.
Speed - percent	Speed percentage.
Speed	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.
manufact - urer	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)	<pre>{   "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" },</pre>
----------------------	--

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)	{ "Serial Number" : "Y052MV4CT00J", }
----------------------	---

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.

---

## Startup Information

The following startup information URIs are available:

- /nos/api/startup GET, PUT
- /nos/api/startup/ztp GET, PUT
- /nos/api/startup/software GET, PUT

The following interface commands are available:

- [Get All Interfaces](#)
- [Put System ZTP Interface](#)
- [Get System ZTP Setting](#)
- [Put System Startup Image](#)
- [Get System Startup Image](#)

## Get System Startup Information

Get 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

Update 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
<i>setting</i>	The ZTP provisioning setting; one of "Enable", "Forcedly Enabled", "Forcedly Disabled".

### Response

Response Body (JSON)	{ "ztp" : "<setting>" }
-------------------------	-------------------------------

where:

Element	Description
<i>setting</i>	Current ZTP provisioning setting.

## Get System ZTP Setting

Get 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
<i>setting</i>	Current ZTP provisioning setting.

## Put System Startup Image

Update 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
<i>setting</i>	Next startup image setting; one of "active" or "standby".

### Response

Response Body (JSON)	{ "boot software" : "<setting>" }
-------------------------	---

where:

Element	Description
<i>setting</i>	Next startup image setting; one of "active" or "standby".

## Get System Startup Image

Get 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
<i>setting</i>	Next startup image setting; one of "active" or "standby".



---

## 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

Get 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
<i>flag</i>	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*

Response Body (JSON)	
-------------------------	--

## Save Configuration

Saves the running configuration to flash memory.

### *Request*

Method Type	GET
Request URI	/nos/api/save/config
Request Body (JSON)	

### *Response*

Response Body (JSON)	
-------------------------	--

## Download Image to Switch

Downloads a boot image to the switch.

### Request

Method Type	POST
Request URI	/nos/api/download/image
Request Body (JSON)	<pre>{   "protocol": "&lt;protocol&gt;",   "serverip": "&lt;serverip&gt;",   "srcfile": "&lt;srcfile&gt;",   "imgtype": "&lt;imgtype&gt;",   "username": "&lt;username&gt;",   "passwd": "&lt;passwd&gt;",   "vrf_name": "&lt;vrf_name&gt;" }</pre>

where:

Element	Description
<i>protocol</i>	Protocol name; one of "tftp", "sftp".
<i>serverip</i>	Server IP address.
<i>srcfile</i>	Source file; up to 256 characters long.
<i>imgtype</i>	System image type; one of "all", "boot", "onie", "os".
<i>username</i>	(Optional) Username for the server.
<i>passwd</i>	(Optional) Password for the server username.
<i>vrf_name</i>	(Optional) VRF name; an alphabetic string up to 64 characters long.

### Response

Response Body (JSON)	<pre>{   "status": "&lt;status&gt;", }</pre>
-------------------------	--

where:

Element	Description
<i>status</i>	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)	<pre>{   "protocol": "&lt;protocol&gt;",   "serverip": "&lt;serverip&gt;",   "srcfile": "&lt;srcfile&gt;",   "dstfile": "&lt;dstfile&gt;",   "username": "&lt;username&gt;",   "passwd": "&lt;passwd&gt;",   "vrf_name": "&lt;vrf_name&gt;" }</pre>

where:

Element	Description
<i>protocol</i>	Protocol name; one of "tftp", "sftp".
<i>serverip</i>	Server IP address.
<i>srcfile</i>	Source file; up to 256 characters long.
<i>dstfile</i>	Destination file; one of "running_config", "startup_config".
<i>username</i>	(Optional) Username for the server.
<i>passwd</i>	(Optional) Password for the server username.
<i>vrf_name</i>	(Optional) VRF name; an alphabetic string up to 64 characters long.

## Response

Response Body (JSON)	<pre>{   "status": "&lt;status&gt;",   "details": "&lt;details&gt;",   "filename": "&lt;filename&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>status</i>	Transfer status; one of "transferring", "installing", "successful", "failed".
<i>details</i>	Detailed description of the status; one of: <ul style="list-style-type: none"><li>• Transferring running-config</li><li>• Transferring startup-config</li><li>• Installing image</li><li>• <i>image</i> installation succeeded</li><li>• Copy success</li><li>• VRF <i>vrf_name</i> doesn't exist</li><li>• Another image installation is in progress</li><li>• Host <i>serverip</i> is unreachable</li><li>• ONIE feature is not enabled on this switch</li><li>• File not found</li><li>• SFTP authentication failure</li><li>• <i>image</i> installation failed</li><li>• Copy failed</li></ul>
<i>filename</i>	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)	<pre>{   "protocol": "&lt;protocol&gt;",   "serverip": "&lt;serverip&gt;",   "srcfile": "&lt;srcfile&gt;",   "dstfile": "&lt;dstfile&gt;",   "username": "&lt;username&gt;",   "passwd": "&lt;passwd&gt;",   "vrf_name": "&lt;vrf_name&gt;" }</pre>

where:

Element	Description
<i>protocol</i>	Protocol name; one of "tftp", "sftp".
<i>serverip</i>	Server IP address.
<i>srcfile</i>	Source file; up to 256 characters long.
<i>dstfile</i>	Destination file; one of "running_config", "startup_config".
<i>username</i>	(Optional) Username for the server.
<i>passwd</i>	(Optional) Password for the server username.
<i>vrf_name</i>	(Optional) VRF name; an alphabetic string up to 64 characters long.



## Response

Response Body (JSON)	<pre>{   "status": "&lt;status&gt;",   "details": "&lt;details&gt;",   "filename": "&lt;filename&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>status</i>	Transfer status; one of "transferring", "installing", "successful", "failed".
<i>details</i>	Detailed description of the status; one of: <ul style="list-style-type: none"><li>• Transferring running-config</li><li>• Transferring startup-config</li><li>• Installing image</li><li>• <i>image</i> installation succeeded</li><li>• Copy success</li><li>• VRF <i>vrf_name</i> doesn't exist</li><li>• Another image installation is in progress</li><li>• Host <i>serverip</i> is unreachable</li><li>• ONIE feature is not enabled on this switch</li><li>• File not found</li><li>• SFTP authentication failure</li><li>• <i>image</i> installation failed</li><li>• Copy failed</li></ul>
<i>filename</i>	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
<i>protocol</i>	Protocol name; one of "tftp", "sftp".
<i>serverip</i>	Server IP address.
<i>dstfile</i>	Destination file; one of "running_config", "startup_config".
<i>username</i>	(Optional) Username for the server.
<i>passwd</i>	(Optional) Password for the server username.
<i>vrf_name</i>	(Optional) VRF name; an alphabetic string up to 64 characters long.

### Response

Response Body (JSON)	{ "status": "<status>", }
-------------------------	---------------------------------

where:

Element	Description
<i>status</i>	Transfer status; one of "transferring", "installing", "successful", "failed".

## Get Download Transfer Status

Get the status of a downloading transfer.

### Request

Method Type	GET
Request URI	/nos/api/download/status/<content>
Request Body (JSON)	

where:

Element	Description
<i>content</i>	One of "image", "running_config", "startup_config".

### Response

Response Body (JSON)	<pre>{   "status": "&lt;status&gt;",   "details": "&lt;details&gt;",   "filename": "&lt;filename&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>status</i>	Transfer status; one of "transferring", "installing", "successful", "failed".
<i>details</i>	Detailed description of the status; one of: <ul style="list-style-type: none"><li>• Transferring running-config</li><li>• Transferring startup-config</li><li>• Installing image</li><li>• <i>image</i> installation succeeded</li><li>• Copy success</li><li>• VRF <i>vrf_name</i> doesn't exist</li><li>• Another image installation is in progress</li><li>• Host <i>serverip</i> is unreachable</li><li>• ONIE feature is not enabled on this switch</li><li>• File not found</li><li>• SFTP authentication failure</li><li>• <i>image</i> installation failed</li><li>• Copy failed</li></ul>
<i>filename</i>	Name of file being downloaded; up to 256 characters long.

## Get Upload Transfer Status

Get the status of an uploading transfer.

### *Request*

Method Type	GET
Request URI	/nos/api/upload/status/<content>
Request Body (JSON)	

where:

Element	Description
<i>content</i>	One of "image", "running_config", "startup_config".

## Response

Response Body (JSON)	<pre>{   "status": "&lt;status&gt;",   "details": "&lt;details&gt;",   "filename": "&lt;filename&gt;" }</pre>
----------------------	---

where:

Element	Description
status	Transfer status; one of "transferring", "installing", "successful", "failed".
details	Detailed description of the status; one of: <ul style="list-style-type: none"><li>• Transferring running-config</li><li>• Transferring startup-config</li><li>• Transferring tech-support</li><li>• Copy success</li><li>• VRF <i>vrf_name</i> doesn't exist</li><li>• Another image installation is in progress</li><li>• Host <i>serverip</i> is unreachable</li><li>• ONIE feature is not enabled on this switch</li><li>• File not found</li><li>• SFTP authentication failure</li><li>• Copy failed</li></ul>
filename	Name of file being saved on the server; up to 256 characters long.

---

## 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 All LAGs](#)
- [Get 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

Get 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
<i>hostname</i>	The hostname of the system; a string from 1-64 characters long.

## Set Hostname

Set 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
<i>hostname</i>	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
<i>date</i>	System date in the format: HH:MM:SS xM ZZZ Wkd Mon Dy YEAR where: <ul style="list-style-type: none"><li>● HH - hour</li><li>● MM - minutes</li><li>● SS - seconds</li><li>● xM - one of "AM", "PM"</li><li>● Wkd - three-letter weekday abbreviation</li><li>● Mon - three-letter month abbreviation</li><li>● Dy - one or two-digit day</li><li>● YEAR - four-digit year</li></ul> 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)	<pre>{   "time": "&lt;HH:MM:SS&gt;" ,   "day": &lt;day&gt;,   "month": &lt;month&gt; ,   "year": &lt;year&gt; }</pre>

where:

Element	Description
time	System time in the 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: <ul style="list-style-type: none"><li>• January</li><li>• February</li><li>• March</li><li>• April</li><li>• May</li><li>• June</li><li>• July</li><li>• August</li><li>• September</li><li>• October</li><li>• November</li><li>• December</li></ul>
year	The year; an integer from 2000-2030.

## Response

Response Body (JSON)	{ "date": "<date>", }
----------------------	-----------------------------

where:

Element	Description
<i>date</i>	System date in the format: HH:MM:SS xM ZZZ Wkd Mon Dy YEAR  where: <ul style="list-style-type: none"><li>● HH - hour</li><li>● MM - minutes</li><li>● SS - seconds</li><li>● xM - one of "AM", "PM"</li><li>● Wkd - three-letter weekday abbreviation</li><li>● Mon - three-letter month abbreviation</li><li>● Dy - one or two-digit day</li><li>● YEAR - four-digit year</li></ul> 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
<i>format</i>	An integer; one of 12, 24.

### *Response*

Response Body (JSON)	{ "format": <format>, }
-------------------------	-------------------------------

where:

Element	Description
<i>format</i>	An integer; one of 12, 24.

## 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	One of "none", "ntp". Default value: "ntp".

### Response

Response Body (JSON)	{ "protocol": "<protocol>", }
-------------------------	-------------------------------------

where:

Element	Description
protocol	One of "none", "ntp".

## Set Clock Timezone

Set 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
<i>timezone</i>	One to five letter string denoting the local system time zone:
<i>offsethour</i>	Hours offset from UTC; an integer from -23 through 23.
<i>offsetmin</i>	Minutes offset from UTC; an integer from 0-59.

### Response

Response Body (JSON)	{ "date": "<date>", }
-------------------------	-----------------------------

where:

Element	Description
<i>date</i>	System date in the format:  HH:MM:SS xM ZZZ Wkd Mon Dy YEAR  where: <ul style="list-style-type: none"><li>● HH - hour</li><li>● MM - minutes</li><li>● SS - seconds</li><li>● xM - one of "AM", "PM"</li><li>● Wkd - three-letter weekday abbreviation</li><li>● Mon - three-letter month abbreviation</li><li>● Dy - one or two-digit day</li><li>● YEAR - four-digit year</li></ul> For example:  10:55:58 AM UTC Mon Jul 4 2016

## Set Clock Summer Time

Set the transition to and from a summer time zone adjustment.

### Request

Method Type	PUT
Request URI	/nos/api/cfg/clock/summertime
Request Body (JSON)	<pre>{   "timezone": &lt;time_zone&gt;,   "startweek": &lt;start_week&gt;,   "startweekday": &lt;start_weekday&gt;,   "startmonth": &lt;start_month&gt;,   "starttime" : "&lt;HH:MM&gt;",   "endweek"   : &lt;end_week&gt;,   "endweekday": &lt;end_weekday&gt;,   "endmonth"  : &lt;end_month&gt;,   "endtime"   : "&lt;HH:MM&gt;",   "offsetmin" : &lt;minutes&gt; }</pre>

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: <ul style="list-style-type: none"><li>● monday</li><li>● tuesday</li><li>● wednesday</li><li>● thursday</li><li>● friday</li><li>● saturday</li><li>● sunday</li></ul>

Element	Description
startmonth	Month to start DST; one of the following case-insensitive strings: <ul style="list-style-type: none"> <li>● january</li> <li>● february</li> <li>● march</li> <li>● april</li> <li>● may</li> <li>● june</li> <li>● july</li> <li>● august</li> <li>● september</li> <li>● october</li> <li>● november</li> <li>● december</li> </ul>
starttime	Time to start DST; a string in the 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: <ul style="list-style-type: none"> <li>● monday</li> <li>● tuesday</li> <li>● wednesday</li> <li>● thursday</li> <li>● friday</li> <li>● saturday</li> <li>● sunday)</li> </ul>
endmonth	Month in which DST ends; one of the following case-insensitive strings: <ul style="list-style-type: none"> <li>● january</li> <li>● february</li> <li>● march</li> <li>● april</li> <li>● may</li> <li>● june</li> <li>● july</li> <li>● august</li> <li>● september</li> <li>● october</li> <li>● november</li> <li>● december</li> </ul>
endtime	Time to end DST; a string in the format "HH:MM"
offsetmin	Offset to add, in minutes; an integer from 1-1440.



## Response

Response Body (JSON)	{ "date": "<date>", }
----------------------	-----------------------------

where:

Element	Description
<i>date</i>	System date in the format: HH:MM:SS xM ZZZ Wkd Mon Dy YEAR  where: <ul style="list-style-type: none"><li>● HH - hour</li><li>● MM - minutes</li><li>● SS - seconds</li><li>● xM - one of "AM", "PM"</li><li>● Wkd - three-letter weekday abbreviation</li><li>● Mon - three-letter month abbreviation</li><li>● Dy - one or two-digit day</li><li>● YEAR - four-digit year</li></ul> For example: 10:55:58 AM UTC Mon Jul 4 2016

## Get Device Contact

Get the device contact.

### Request

Method Type	GET
Request URI	/nos/api/cfg/contact
Request Body (JSON)	{ "contact": <contact>, }

where:

Element	Description
<i>contact</i>	Device contact; a string up to 256 characters long.

### Response

Response Body (JSON)	{ "contact": <contact>, }
-------------------------	---------------------------------

where:

Element	Description
<i>contact</i>	Device contact; a string up to 256 characters long.

## Update Device Contact

Update the device contact.

### *Request*

Method Type	PUT
Request URI	/nos/api/cfg/contact
Request Body (JSON)	{ "contact": <contact>, }

where:

Element	Description
<i>contact</i>	Device contact; a string up to 256 characters long.

### *Response*

Response Body (JSON)	{ "contact": <contact>, }
-------------------------	---------------------------------

where:

Element	Description
<i>contact</i>	Device contact; a string up to 256 characters long.

## Get Device Description

Get the device description.

### Request

Method Type	GET
Request URI	/nos/api/cfg/descr
Request Body (JSON)	{ "descr": <descr>, }

where:

Element	Description
<i>contact</i>	Device description; a string up to 256 characters long.

### Response

Response Body (JSON)	{ "descr": <descr>, }
-------------------------	-----------------------------

where:

Element	Description
<i>contact</i>	Device description; a string up to 256 characters long.

## Update Device Description

Update the device description.

### *Request*

Method Type	PUT
Request URI	/nos/api/cfg/descr
Request Body (JSON)	{ "descr": <descr>, }

where:

Element	Description
<i>contact</i>	Device description; a string up to 256 characters long.

### *Response*

Response Body (JSON)	{ "descr": <descr>, }
-------------------------	-----------------------------

where:

Element	Description
<i>contact</i>	Device description; a string up to 256 characters long.

## Get Rack Properties

Get the rack properties for the switch.

### Request

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

### Response

Response Body (JSON)	<pre>{   "location": "&lt;location&gt;",   "room": "&lt;room&gt;",   "rack": "&lt;rack&gt;",   "lru": "&lt;lru&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>location</i>	Device location; a string up to 256 characters long.
<i>room</i>	Device room ID; a string up to 256 characters long.
<i>rack</i>	Device Rack; a string up to 256 characters long.
<i>lru</i>	Device lowest rack unit; a string up to 256 characters long.

## Update Rack Properties

Update the rack properties for the switch.

### Request

Method Type	PUT
Request URI	/nos/api/cfg/rack_prop
Request Body (JSON)	<pre>{   "location": "&lt;location&gt;",   "room": "&lt;room&gt;",   "rack": "&lt;rack&gt;",   "lru": "&lt;lru&gt;" }</pre>

where:

Element	Description
<i>location</i>	(Optional) Device location; a string up to 256 characters long.
<i>room</i>	(Optional) Device room ID; a string up to 256 characters long.
<i>rack</i>	(Optional) Device Rack; a string up to 256 characters long.
<i>lru</i>	(Optional) Device lowest rack unit; a string up to 256 characters long.

### Response

Response Body (JSON)	<pre>{   "location": "&lt;location&gt;",   "room": "&lt;room&gt;",   "rack": "&lt;rack&gt;",   "lru": "&lt;lru&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>location</i>	Device location; a string up to 256 characters long.
<i>room</i>	Device room ID; a string up to 256 characters long.
<i>rack</i>	Device Rack; a string up to 256 characters long.
<i>lru</i>	Device lowest rack unit; a string up to 256 characters long.

---

## 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

Get properties of all interfaces.

### Request

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

### Response

Response Body (JSON)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "duplex": "&lt;duplex&gt;",     "speed": "&lt;speed&gt;",     "mtu": "&lt;mtu&gt;",     "mac_addr": "&lt;mac_addr&gt;",     "admin_state": "&lt;admin_state&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	The interface name; a string. <b>Note:</b> The interface must exist.
<i>duplex</i>	The communication method of the interface; one of <code>auto</code> , <code>full</code> , <code>half</code> .
<i>speed</i>	The communication speed of the interface; one of the following: <ul style="list-style-type: none"><li>● <code>auto</code> (auto negotiate)</li><li>● <code>10</code> (10Mb/s)</li><li>● <code>100</code> (100Mb/s)</li><li>● <code>1000</code> (1Gb/s)</li><li>● <code>10000</code> (10Gb/s)</li><li>● <code>40000</code> (40Gb/s).</li></ul>
<i>mtu</i>	The maximum transmission unit, in bytes; a positive integer from 64-9216.
<i>mac_addr</i>	The MAC address (in <code>xxxx.xxxx.xxxx</code> format).
<i>admin_state</i>	The admin status; one of <code>up</code> , <code>down</code> .

## Get Interface

Get properties of one interface.

### Request

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

where:

Element	Description
<i>if_name</i>	The interface name (Str). <b>Note:</b> The interface must exist.

### Response

Response Body  (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "duplex": "&lt;duplex&gt;",   "speed": "&lt;speed&gt;",   "mtu": {mtu},   "mac_addr": "&lt;mac_addr&gt;",   "admin_state": "&lt;admin_state&gt;" }</pre>
--------------------------------	---

where:

Element	Description
<i>if_name</i>	The interface name (Str).
<i>duplex</i>	The communication method of the interface; auto, full, half.
<i>speed</i>	The communication speed of the interface; auto (auto negotiate), 10 (10Mb/s), 100 (100Mb/s), 1000 (1Gb/s), 10000 (10Gb/s), 40000 (40Gb/s).
<i>mtu</i>	The maximum transmission unit, in bytes; a positive integer from 64-9216.
<i>mac_addr</i>	The MAC address (in xxxx.xxxx,xxxx format).
<i>admin_state</i>	The admin status; up, down.

## Update Interface

Update 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
<i>if_name</i>	The interface name (Str). <b>Note:</b> The interface must exist.
<i>mtu</i>	The maximum transmission unit, in bytes; a positive integer from 64-9216. Default value: 1500.
<i>admin_state</i>	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)	{ "if_name": "<if_name>", "duplex": "<duplex>", "speed": "<speed>", "mtu": "<mtu>", "mac_addr": "<mac_addr>", "admin_state": "<admin_state>" }
-------------------------	---

where:

Element	Description
<i>if_name</i>	The interface name (Str).
<i>duplex</i>	The communication method of the interface; auto, full, half.
<i>speed</i>	The communication speed of the interface; auto (auto negotiate), 10 (10Mb/s), 100 (100Mb/s), 1000 (1Gb/s), 10000 (10Gb/s), 40000 (40Gb/s).

Element	Description
mtu	The maximum transmission unit, in bytes; a positive integer from 64-9216.
mac_addr	The MAC address (in xxxx.xxxx,xxxx format).
admin_state	The admin status; up, down.

### Example

Method Type	PUT
Request URI	/nos/api/cfg/interface/Ethernet1%2F5
Request Body (JSON)	<pre>{   "if_name": "Ethernet1/5",   "duplex": "duplex-full",   "speed": "auto",   "mtu": 9216,   "mac_addr": "0001.0200.0005",   "admin_state": "up" }</pre>

## Get Transceiver Information for All Interfaces

Get transceiver information for all interfaces.

### Request

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

### Response

Response Body (JSON)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "installed": "&lt;installed&gt;",     "status": "&lt;status&gt;",     "type": "&lt;type&gt;",     "vendor": "&lt;vendor&gt;",     "part number": "&lt;part_number&gt;",     "revision": "&lt;revision&gt;",     "serial number": "&lt;serial_number&gt;",     "volts": "&lt;volts&gt;",     "temperature": "&lt;temperature&gt;",     "approved": "&lt;approved&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>installed</i>	Whether the transceiver is present; one of present, not present.
<i>status</i>	Interface status; one of enabled, disabled.
<i>type</i>	Type of transceiver; a string.
<i>vendor</i>	Vendor of transceiver; a string.
<i>part_number</i>	Part number of transceiver; a string.
<i>revision</i>	Revision of transceiver; a string or N/A.
<i>serial_number</i>	Serial number of transceiver; a string.
<i>volts</i>	Volts of transceiver; a string or N/A.

<b>Element</b>	<b>Description</b>
<i>temperature</i>	Temperature of transceiver; a string or N/A.
<i>approved</i>	Approval status of the transceiver; one of approved, accepted, unapproved, Unsuport, Restrict, No Device.

## Get Transceiver Information for One Interfaces

Get 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
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.

### Response

Response Body (JSON)	[ { "if_name": "<if_name>", "installed": "<installed>", "status": "<status>", "type": "<type>", "vendor": "<vendor>", "part number": "<part_number>", "revision": "<revision>", "serial number": "<serial_number>", "volts": "<volts>", "temperature": "<temperature>", "approved": "<approved>" } ]
-------------------------	--

where:

Element	Description
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>installed</i>	Whether the transceiver is present; one of present, not present.
<i>status</i>	Interface status; one of enabled, disabled.
<i>type</i>	Type of transceiver; a string.
<i>vendor</i>	Vendor of transceiver; a string.
<i>part_number</i>	Part number of transceiver; a string.

<b>Element</b>	<b>Description</b>
<i>revision</i>	Revision of transceiver; a string or N/A.
<i>serial number</i>	Serial number of transceiver; a string.
<i>volts</i>	Volts of transceiver; a string or N/A.
<i>temperature</i>	Temperature of transceiver; a string or N/A.
<i>approved</i>	Approval status of the transceiver; one of approved, accepted, unapproved, Unsuport, Restrict, No Device.



---

## LACP

The following LACP URIs are 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

Get the LACP properties of the system.

### Request

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

### Response

Response Body (JSON)	<pre>{   "sys_prio": "&lt;sys_prio&gt;",   "max_bundle": "&lt;max_bundle&gt;",   "interfaces": [     {       "if_name": "&lt;if_name&gt;",       "lag_mode": "&lt;lag_mode&gt;",       "lacp_prio": "&lt;lacp_prio&gt;",       "lacp_timeout": "&lt;lacp_timeout&gt;"     }   ] }</pre>
-------------------------	---

where:

Element	Description
<i>sys_prio</i>	LACP system priority; a positive integer from 1-65535. Default value: 32768.
<i>max_bundle</i>	The supported maximum number of links per LAG; a positive integer.
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The interface must exist.
<i>lag_mode</i>	LAG mode; one of <code>lacp_active</code> , <code>lacp_passive</code> , <code>no_lacp</code>
<i>lacp_prio</i>	LACP priority for the physical port a positive integer from 1-65535. Default value: 32768.
<i>lacp_timeout</i>	LACP timeout for the physical port; one of <code>short</code> , <code>long</code> (default).

## Update LACP System Properties

Update 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
<i>sys_prio</i>	LACP system priority; a positive integer from 1-65535. Default value: 32768.

### Response

Response Body (JSON)	{ "sys_prio": "<sys_prio>", }
-------------------------	-------------------------------------

where:

Element	Description
<i>sys_prio</i>	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, DELETE
- `/nos/api/cfg/lag/<lag_id>` GET, PUT, DELETE

The following LAG commands are available:

- [Get All LAGs](#)
- [Create LAG](#)
- [Get LAG Properties](#)
- [Update LAG](#)
- [Delete LAG](#)

## Get All LAGs

Get properties of all LAGs.

### Request

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

### Response

Response Body (JSON)	<pre>[   {     "lag_name": "&lt;lag_name&gt;",     "lag_id": "&lt;lag_id&gt;",     "interfaces": [       {         "if_name": "&lt;if_name&gt;",         "lag_mode": "&lt;lag_mode&gt;",         "lACP_prio": "&lt;lACP_prio&gt;",         "lACP_timeout": "&lt;lACP_timeout&gt;"       }     ],     "suspend_individual": "&lt;status&gt;",   } ]</pre>
-------------------------	--

where:

Element	Description
<i>lag_name</i>	The name of the LAG; a string.
<i>lag_id</i>	LAG identifier; an integer from 1-65535
<i>interfaces</i>	Physical interface members of the LAG; an integer from 1-32.
<i>if_name</i>	Ethernet interface name. <b>Note:</b> The interface must exist.
<i>lag_mode</i>	LAG mode; one of <code>lACP_active</code> , <code>lACP_passive</code> , <code>no_lACP</code> .
<i>lACP_prio</i>	LACP priority for the physical port; an integer from 1-65535. Default value: 32768.
<i>lACP_timeout</i>	LACP timeout for the physical port; one of <code>short</code> , <code>long</code> . Default value: <code>long</code> .

Element	Description
<i>suspend_individual</i>	<p>If the LAG does not get the LACP BPUD from peer ports the port aggregation, the result is one of the following:</p> <ul style="list-style-type: none"> <li>● <b>Yes</b> - LACP on the the ports is suspended rather than put into individual state.</li> <li>● <b>No</b>: LAG on the ports is put into individual state.</li> </ul> <p>Default value: NO.</p>

## Create LAG

Creates a LAG.

### Request

Method Type	POST
Request URI	/nos/api/cfg/lag
Request Body (JSON)	<pre>{   "lag_id": "&lt;lag_id&gt;",   "interfaces": [     {       "if_name": "&lt;if_name&gt;",       "lag_mode": "&lt;lag_mode&gt;",       "lACP_prio": "&lt;lACP_prio&gt;",       "lACP_timeout": "&lt;lACP_timeout&gt;"     }   ] }</pre>

where:

Element	Description
<i>lag_id</i>	LAG identifier; a positive integer from 1-4096.
<i>interfaces</i>	Physical interface members of the LAG. Up to 32 interfaces can be added.
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The interface must exist.
<i>lag_mode</i>	LAG mode; one of <code>lACP_active</code> , <code>lACP_passive</code> , <code>no_lACP</code> .
<i>lACP_prio</i>	(Optional) LACP priority for the physical port; a positive integer from 1-65535. Default value: 32768.
<i>lACP_timeout</i>	(Optional) LACP timeout for the physical port; one of <code>short</code> , <code>long</code> . Default value: <code>long</code> .

## Response

Response Body (JSON)	<pre>{   "lag_id": "&lt;lag_id&gt;",   "lag_name": "&lt;lag_name&gt;",   "interfaces": [     {       "if_name": "&lt;if_name&gt;",       "lag_mode": "&lt;lag_mode&gt;",       "lACP_prio": "&lt;lACP_prio&gt;",       "lACP_timeout": "&lt;lACP_timeout&gt;"     }   ] }</pre>
----------------------	---

where:

Element	Description
<i>lag_name</i>	LAG name.
<i>lag_id</i>	LAG identifier; a positive integer from 1-4096.
<i>interfaces</i>	Physical interface members of the LAG. Up to 32 interfaces can be added.
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The interface must exist.
<i>lag_mode</i>	LAG mode; one of <code>lACP_active</code> , <code>lACP_passive</code> , <code>no_lACP</code> .
<i>lACP_prio</i>	LACP priority for the physical port; a positive integer from 1-65535. Default value: 32768.
<i>lACP_timeout</i>	LACP timeout for the physical port; one of <code>short</code> , <code>long</code> . Default value: <code>long</code> .



## Get LAG Properties

Get properties of the specified LAG.

### Request

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

where:

Element	Description
<i>lag_id</i>	LAG identifier; an integer from 1-65535.

### Response

Response Body (JSON)	<pre>[   {     "lag_name": "&lt;lag_name&gt;",     "lag_id": "&lt;lag_id&gt;",     "interfaces": [       {         "if_name": "&lt;if_name&gt;",         "lag_mode": "&lt;lag_mode&gt;",         "lACP_prio": "&lt;lACP_prio&gt;",         "lACP_timeout": "&lt;lACP_timeout&gt;"       }     ],     "suspend_individual": "&lt;status&gt;",   } ]</pre>
-------------------------	--

where:

Element	Description
<i>lag_name</i>	The name of the LAG; a string.
<i>lag_id</i>	LAG identifier; an integer from 1-65535
<i>interfaces</i>	Physical interface members of the LAG; an integer from 1-32.
<i>if_name</i>	Ethernet interface name. <b>Note:</b> The interface must exist.
<i>lag_mode</i>	LAG mode; one of <code>lACP_active</code> , <code>lACP_passive</code> , <code>no_lACP</code> .
<i>lACP_prio</i>	LACP priority for the physical port; an integer from 1-65535. Default value: 32768.

Element	Description
<i>lacp_timeout</i>	LACP timeout for the physical port; one of <code>short</code> , <code>long</code> . Default value: <code>long</code> .
<i>suspend_individual</i>	<p>If the LAG does not get the LACP BPUD from peer ports the port aggregation, the result is one of the following:</p> <ul style="list-style-type: none"> <li>● <b>Yes</b> - LACP on the the ports is suspended rather than put into individual state.</li> <li>● <b>No</b>: LAG on the ports is put into individual state.</li> </ul> <p>Default value: <b>No</b>.</p>

## 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)	<pre>{   "interfaces": [     {       "if_name": "&lt;if_name&gt;",       "lag_mode": "&lt;lag_mode&gt;",       "lACP_prio": "&lt;lACP_prio&gt;",       "lACP_timeout": "&lt;lACP_timeout&gt;"     }   ],   "suspend_individual": "&lt;status&gt;", }</pre>

where:

Element	Description
<i>lag_id</i>	LAG identifier; an integer from 1-65535
<i>interfaces</i>	(Optional) Physical interface members of the LAG; an integer from 1-32.
<i>if_name</i>	Ethernet interface name. <b>Note:</b> The interface must exist.
<i>lag_mode</i>	LAG mode; one of <code>lACP_active</code> , <code>lACP_passive</code> , <code>no_lACP</code> .
<i>lACP_prio</i>	(Optional) LACP priority for the physical port; an integer from 1-65535. Default value: 32768.
<i>lACP_timeout</i>	(Optional) LACP timeout for the physical port; one of <code>short</code> , <code>long</code> . Default value: <code>long</code> .
<i>suspend_individual</i>	(Optional) If the LAG does not get the LACP BPUD from peer ports the port aggregation, the result is one of the following: <ul style="list-style-type: none"> <li>• <b>Yes</b> - LACP on the the ports is suspended rather than put into individual state.</li> <li>• <b>No</b>: LAG on the ports is put into individual state.</li> </ul> Default value: <code>No</code> .

## Response

Response Body (JSON)	<pre>[   {     "lag_id": "&lt;lag_id&gt;",     "lag_name": "&lt;lag_name&gt;",     "interfaces": [       {         "if_name": "&lt;if_name&gt;",         "lag_mode": "&lt;lag_mode&gt;",         "lACP_prio": "&lt;lACP_prio&gt;",         "lACP_timeout": "&lt;lACP_timeout&gt;"       }     ],     "suspend_individual": "&lt;status&gt;",   } ]</pre>
-------------------------	--

where:

Element	Description
<i>lag_id</i>	LAG identifier; an integer from 1-65535
<i>lag_name</i>	The name of the LAG; a string.
<i>interfaces</i>	Physical interface members of the LAG; an integer from 1-32.
<i>if_name</i>	Ethernet interface name. <b>Note:</b> The interface must exist.
<i>lag_mode</i>	LAG mode; one of <code>lACP_active</code> , <code>lACP_passive</code> , <code>no_lACP</code> .
<i>lACP_prio</i>	LACP priority for the physical port; an integer from 1-65535. Default value: 32768.
<i>lACP_timeout</i>	LACP timeout for the physical port; one of <code>short</code> , <code>long</code> . Default value: <code>long</code> .
<i>suspend_individual</i>	If the LAG does not get the LACP BPUD from peer ports the port aggregation, the result is one of the following: <ul style="list-style-type: none"> <li>● <b>Yes</b> - LACP on the the ports is suspended rather than put into individual state.</li> <li>● <b>No</b>: LAG on the ports is put into individual state.</li> </ul> Default value: <code>No</code> .

## Get LAG Load Balance Settings

Get 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
<i>destination-ip</i>	Load distribution on the destination IP address.
<i>destination-mac</i>	Load distribution on the destination MAC address.
<i>destination-port</i>	Load distribution on the destination TCP/UDP port.
<i>source-dest-ip</i>	Load distribution on the source and destination IP address.
<i>source-dest-mac</i>	Load distribution on the source and destination MAC address.
<i>source-dest-port</i>	Load distribution on the source and destination TCP/UDP port.
<i>source-interface</i>	Load distribution on the source ethernet interface.
<i>source-ip</i>	Load distribution on the source IP address.
<i>source-mac</i>	Load distribution on the source MAC address.
<i>source-port</i>	Load distribution on the source TCP/UDP port.

## Update LAG Load Balance Settings

Update 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
<i>destination-ip</i>	Load distribution on the destination IP address.
<i>destination-mac</i>	Load distribution on the destination MAC address.
<i>destination-port</i>	Load distribution on the destination TCP/UDP port.
<i>source-dest-ip</i>	Load distribution on the source and destination IP address.
<i>source-dest-mac</i>	Load distribution on the source and destination MAC address.
<i>source-dest-port</i>	Load distribution on the source and destination TCP/UDP port.
<i>source-interface</i>	Load distribution on the source ethernet interface.
<i>source-ip</i>	Load distribution on the source IP address.
<i>source-mac</i>	Load distribution on the source MAC address.
<i>source-port</i>	Load distribution on the source TCP/UDP port.

## 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
<i>destination-ip</i>	Load distribution on the destination IP address.
<i>destination-mac</i>	Load distribution on the destination MAC address.
<i>destination-port</i>	Load distribution on the destination TCP/UDP port.
<i>source-dest-ip</i>	Load distribution on the source and destination IP address.
<i>source-dest-mac</i>	Load distribution on the source and destination MAC address.
<i>source-dest-port</i>	Load distribution on the source and destination TCP/UDP port.
<i>source-interface</i>	Load distribution on the source ethernet interface.
<i>source-ip</i>	Load distribution on the source IP address.
<i>source-mac</i>	Load distribution on the source MAC address.
<i>source-port</i>	Load distribution on the source TCP/UDP port.

## Delete LAG

Delete a LAG.

### *Request*

Method Type	DELETE
Request URI	/nos/api/cfg/lag/<lag_id>
Request Body (JSON)	

where:

Element	Description
<i>lag_id</i>	LAG identifier; a positive integer from 1-4096.

**Note:** If there is no *lag\_id* (*lag\_id*=None or specified *lag\_id* is "all"), all user-created LAGs will be deleted.



---

## 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
- /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

Get 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)	<pre>{   "status": "&lt;status&gt;",   "tier_id": "&lt;tier_id&gt;",   "priority": "&lt;priority&gt;",   "auto_recover" : "&lt;auto_recover&gt;",   "startup_delay": "&lt;startup_delay&gt;", }</pre>
-------------------------	---

where:

Element	Description
<i>status</i>	Whether the vLAG is enabled or disabled; one of enable, disable. Default value; disable
<i>tier_id</i>	vLAG tier ID value; an integer from 1-512. Default value: 0.
<i>priority</i>	vLAG priority value; an integer from 0-65535. Default value: 0.
<i>auto_recover</i>	Time interval, in seconds; an integer from 240-3600. Default value: 300.
<i>startup_delay</i>	Delay time, in seconds; an integer from 0-3600. Default value: 120.

## Update vLAG Configuration

Update the Virtual Link Aggregation Group (vLAG) global configuration.

### Request

Method Type	PUT
Request URI	/nos/api/cfg/vlag
Request Body (JSON)	<pre>{   "status": "&lt;status&gt;",   "tier_id": "&lt;tier_id&gt;",   "priority": "&lt;priority&gt;",   "auto_recover" : "&lt;auto_recover&gt;",   "startup_delay": "&lt;startup_delay&gt;", }</pre>

where:

Element	Description
<i>status</i>	Whether the vLAG is enabled or disabled; one of enable, disable. Default value; disable
<i>tier_id</i>	vLAG tier ID value; an integer from 1-512. Default value: 0.
<i>priority</i>	vLAG priority value; an integer from 0-65535. Default value: 0.
<i>auto_recover</i>	Time interval, in seconds; an integer from 240-3600. Default value: 300.
<i>startup_delay</i>	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)	<pre>{   "status": "&lt;status&gt;",   "tier_id": "&lt;tier_id&gt;",   "priority": "&lt;priority&gt;",   "auto_recover" : "&lt;auto_recover&gt;",   "startup_delay": "&lt;startup_delay&gt;", }</pre>
-------------------------	---

where:

Element	Description
<i>status</i>	Whether the vLAG is enabled or disabled; one of enable, disable. Default value; disable
<i>tier_id</i>	vLAG tier ID value; an integer from 1-512. Default value: 0.
<i>priority</i>	vLAG priority value; an integer from 0-65535. Default value: 0.

<b>Element</b>	<b>Description</b>
<i>auto_recover</i>	Time interval, in seconds; an integer from 240-3600. Default value: 300.
<i>startup_delay</i>	Delay time, in seconds; an integer from 0-3600. Default value: 120.

## Get Global vLAG Information

Get global vLAG information.

### Request

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

### Response

Response Body (JSON)	<pre>{   "status": "&lt;status&gt;",   "system_mac": "&lt;system_mac&gt;",   "fdb_refresh": "&lt;fdb_refresh&gt;",   "fdb_synch": "&lt;fdb_synch&gt;",   "auto_recovery":   {     "interval": "&lt;interval&gt;"     "state": "&lt;state&gt;",   }   "startup_delay":   {     "interval": "&lt;interval&gt;"     "state": "&lt;state&gt;",   }   "local":   {     "tier_id": "&lt;tier_id&gt;",     "sys_type": "&lt;sys_type&gt;",     "os_version": "&lt;os_version&gt;",     "admin_role": "&lt;admin_role&gt;",     "oper_role": "&lt;oper_role&gt;",     "priority": "&lt;priority&gt;",     "system_mac": "&lt;system_mac&gt;",     "match": "&lt;match&gt;"   }   "peer":   {     "tier_id": "&lt;tier_id&gt;",     "sys_type": "&lt;sys_type&gt;",     "os_version": "&lt;os_version&gt;",     "admin_role": "&lt;admin_role&gt;",     "oper_role": "&lt;oper_role&gt;",     "priority": "&lt;priority&gt;",     "system_mac": "&lt;system_mac&gt;",     "match": "&lt;match&gt;"   } }</pre>
-------------------------	---

where:

Element	Description
<i>status</i>	Whether the vLAG is enabled or disabled; one of <i>enable</i> , <i>disable</i> . Default value: <i>disable</i> .
<i>system_mac</i>	Unique vLAG system MAC used for LACP negotiation on the vLAG ports so the access switch forms a single LAG. The vLAG <i>tier_id</i> is used to form this vLAG system MAC.
<i>fdb_refresh</i>	Whether FDB refresh is configured; one of <i>yes</i> , <i>no</i> .
<i>fdb_synch</i>	Whether FDB is synchronized; one of <i>yes</i> , <i>no</i> .
<i>auto_recovery</i>	A dictionary consisting of the following values: <ul style="list-style-type: none"> <li>• <i>interval</i>: Time interval, in seconds; an integer from 240-3600. Default value: 300.</li> <li>• <i>state</i>: Auto-recovery state; one of <i>unstarted</i>, <i>running</i>, <i>finished</i>.</li> </ul>
<i>startup_delay</i>	A dictionary consisting of the following values: <ul style="list-style-type: none"> <li>• <i>interval</i>: Delay time, in seconds; an integer from 0-3600. Default value: 120.</li> <li>• <i>state</i>: Startup delay state; one of <i>unstarted</i>, <i>running</i>, <i>finished</i>.</li> </ul>
<i>local</i>	Dictionary containing the following values: <ul style="list-style-type: none"> <li>• <i>tier_id</i>: vLAG tier ID of the local switch.</li> <li>• <i>sys_type</i>: Lenovo hardware model number.</li> <li>• <i>os_version</i>: CNOS version.</li> <li>• <i>admin_role</i>: One of <i>Primary</i>, <i>Secondary</i>, <i>Unselected</i>.</li> <li>• <i>oper_role</i>: One of <i>Primary</i>, <i>Secondary</i>, <i>Unselected</i>,</li> <li>• <i>priority</i>: The local vLAG priority</li> <li>• <i>system_mac</i>: Local switch MAC.</li> <li>• <i>match</i>: Whether there is an ISL local match or mismatch; one of <i>Match</i>, <i>Mis-Match</i>.</li> </ul>
<i>peer</i>	Dictionary containing the following values: <ul style="list-style-type: none"> <li>• <i>tier_id</i>: vLAG tier ID of the peer switch.</li> <li>• <i>sys_type</i>: Lenovo hardware model number.</li> <li>• <i>os_version</i>: CNOS version.</li> <li>• <i>admin_role</i>: One of <i>Primary</i>, <i>Secondary</i>, <i>Unselected</i>.</li> <li>• <i>oper_role</i>: One of <i>Primary</i>, <i>Secondary</i>, <i>Unselected</i>,</li> <li>• <i>priority</i>: The peer vLAG priority</li> <li>• <i>system_mac</i>: Peer switch MAC.</li> <li>• <i>match</i>: Whether there is an ISL local match or mismatch; one of <i>Match</i>, <i>Mis-Match</i>.</li> </ul>

## Get vLAG ISL Information

Get 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
<i>port_aggregator</i>	LAG identifier; an integer from 1-4096.
<i>if_index</i>	ISL interface index.
<i>state</i>	ISL state; one of Down, Inactive, Active.
<i>prev_state</i>	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)	

### *Response*

True if the operation succeeded; otherwise False.



## Get vLAG Health Check Information

Get 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" : "&lt;status&gt;",   "peer_ip": "&lt;peer_ip&gt;",   "vrf": "&lt;vrf&gt;",   "local_ip": "&lt;local_ip&gt;",   "retry_interval": "&lt;retry_interval&gt;",   "keepalive_attempts" : "&lt;keepalive_attempts&gt;",   "keepalive_interval" : "&lt;keepalive_interval&gt;", }</pre>
-------------------------	---

where:

Element	Description
<i>status</i>	vLAG health check status; one of up, down.
<i>peer_ip</i>	IP address of peer switch. This can be the management IP address of the peer switch.
<i>vrf</i>	VRF context string.
<i>local_ip</i>	IP address of local switch. This can be the management IP address of the local switch.
<i>retry_interval</i>	Time interval, in seconds; an integer from 1-300. Default value: 30.
<i>keepalive_attempts</i>	Number of keepalive attempts made before declaring the peer is down; an integer from 1-24. Default value: 3.
<i>keepalive_interval</i>	Time interval, in seconds; an integer from 2-300. Default value: 5.

## Configure vLAG Health Check Parameters

Configure vLAG health check parameters.

### Request

Method Type	PUT
Request URI	/nos/api/cfg/vlag/health_check
Request Body (JSON)	<pre>{   "peer_ip": "&lt;peer_ip&gt;",   "vrf": "&lt;vrf&gt;",   "retry_interval": "&lt;retry_interval&gt;",   "keepalive_attempts" : "&lt;keepalive_attempts&gt;",   "keepalive_interval" : "&lt;keepalive_interval&gt;", }</pre>

where:

Element	Description
<i>peer_ip</i>	IP address of peer switch. This can be the management IP address of the peer switch.
<i>vrf</i>	VRF context string.
<i>retry_interval</i>	Time interval, in seconds; an integer from 1-300. Default value: 30.
<i>keepalive_attempts</i>	Number of keepalive attempts made before declaring the peer is down; an integer from 1-24. Default value: 3.
<i>keepalive_interval</i>	Time interval, in seconds; an integer from 2-300. Default value: 5.

### Response

Response Body (JSON)	<pre>{   "peer_ip": "&lt;peer_ip&gt;",   "vrf": "&lt;vrf&gt;",   "retry_interval": "&lt;retry_interval&gt;",   "keepalive_attempts" : "&lt;keepalive_attempts&gt;",   "keepalive_interval" : "&lt;keepalive_interval&gt;", }</pre>
-------------------------	--

where:

Element	Description
<i>peer_ip</i>	IP address of peer switch. This can be the management IP address of the peer switch.
<i>vrf</i>	VRF context string.
<i>retry_interval</i>	Time interval, in seconds; an integer from 1-300. Default value: 30.

<b>Element</b>	<b>Description</b>
<i>keepalive_attempts</i>	Number of keepalive attempts made before declaring the peer is down; an integer from 1-24. Default value: 3.
<i>keepalive_interval</i>	Time interval, in seconds; an integer from 2-300. Default value: 5.

## Create vLAG Instance

Create a Virtual Link Aggregation Group (vLAG) instance.

### Request

Method Type	POST
Request URI	/nos/api/cfg/vlag/instance
Request Body (JSON)	{ "instance_id": "<instance_id>", "port_aggregator": "<port_aggregator>", "status": "<status>", }

where:

Element	Description
<i>instance_id</i>	vLAG instance ID number; an integer from 1-64.
<i>port_aggregator</i>	LAG identifier; an integer from 1-4096.
<i>status</i>	vLAG status; one of enable, disable. Default value: disable.

### Response

Response Body (JSON)	{ "instance_id": "<instance_id>", "port_aggregator": "<port_aggregator>", "status": "<status>", }
-------------------------	---

where:

Element	Description
<i>instance_id</i>	vLAG instance ID number; an integer from 1-64.
<i>port_aggregator</i>	LAG identifier; an integer from 1-4096.
<i>status</i>	vLAG status; one of enable, disable. Default value: disable.

## Update vLAG Instance

Update 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>",&br/>  "status": "<status>",&br/>}

where:

Element	Description
<i>instance_id</i>	vLAG instance ID number; an integer from 1-64.
<i>port_aggregator</i>	LAG identifier; an integer from 1-4096.
<i>status</i>	vLAG status; one of <code>enable</code> , <code>disable</code> . Default value: <code>disable</code> .

### Response

Response Body (JSON)	{ "instance_id": "<instance_id>",&br/>  "port_aggregator": "<port_aggregator>",&br/>  "status": "<status>",&br/>}
-------------------------	--

where:

Element	Description
<i>instance_id</i>	vLAG instance ID number; an integer from 1-64.
<i>port_aggregator</i>	LAG identifier; an integer from 1-4096.
<i>status</i>	vLAG status; one of <code>enable</code> , <code>disable</code> . Default value: <code>disable</code> .

## Delete vLAG Instance

Delete a vLAG instance.

### *Request*

Method Type	DELETE
Request URI	/nos/api/cfg/vlag/instance/<instance_id>
Request Body (JSON)	

where:

Element	Description
<i>instance_id</i>	vLAG instance ID number; an integer from 1-64.

### *Response*

True if the operation succeeded; otherwise False.

## Get vLAG Instance Configuration

Get 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
<i>instance_id</i>	vLAG instance ID; either <b>NONE</b> or an integer from 1-64.

### Response

Response Body (JSON)	{ "instance_id": "<instance_id>", "port_aggregator": "<port_aggregator>", "status": "<status>", }
-------------------------	---

where:

Element	Description
<i>instance_id</i>	vLAG instance ID number; an integer from 1-64.
<i>port_aggregator</i>	LAG identifier; an integer from 1-4096.
<i>status</i>	vLAG status; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .

## Get vLAG Instance Information

Get 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
<i>instance_id</i>	vLAG instance ID; either <code>NONE</code> or an integer from 1-64.

### Response

Response Body (JSON)	{ "port_aggregator": "<port_aggregator>", "if_index": "<if_index>", "state": "<state>", "prev_state" : "<prev_state>", }
-------------------------	---

where:

Element	Description
<i>port_aggregator</i>	LAG identifier; an integer from 1-4096.
<i>if_index</i>	ISL interface index.
<i>state</i>	ISL state; one of <code>Down</code> , <code>Inactive</code> , <code>Active</code> .
<i>prev_state</i>	Previous ISL state; one of <code>Down</code> , <code>Inactive</code> , <code>Active</code> .



---

## VLAN

The following VLAN-related URIs are available:

- /nos/api/cfg/vlan GET, POST
- /nos/api/cfg/vlan/<vlan\_id> GET, PUT, DELETE

The following VLAN commands are available:

- [Get All VLANs](#)
- [Create VLAN](#)
- [Get VLAN](#)
- [Delete VLAN](#)

## Get All VLANs

Get properties of all VLANs.

### Request

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

### Response

Response Body (JSON)	<pre>[   {     "vlan_name": "&lt;vlan_name&gt;",     "vlan_id": "&lt;vlan_id&gt;",     "admin_state": "&lt;admin_state&gt;",     "mst_inst_id": "&lt;mst_inst_id&gt;",     "interfaces": [       {         "if_name": "&lt;if_name&gt;",         "bridge_port_mode": "&lt;bridge_port_mode&gt;",         "pvid": "&lt;pvid&gt;"       }     ]   } ]</pre>
-------------------------	---

where:

Element	Description
<i>vlan_name</i>	The name of the VLAN.
<i>vlan_id</i>	VLAN number.; an integer from 2-3999.
<i>admin_state</i>	The admin status; one of up, down.
<i>mst_inst_id</i>	MST instance ID; an integer from 0-64. Default value: 0. <b>Note:</b> Instance 0 refers to the CIST.
<i>interfaces</i>	Interface members of a VLAN. <b>Note:</b> The interface members must exist.
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The ethernet interface must exist.

## Create VLAN

Create 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
<i>vlan_name</i>	VLAN name; a string up to 32 characters long. To create a vlan with the default name, the <i>vlan_name</i> field must be null.
<i>vlan_id</i>	VLAN number.; an integer from 2-3999.
<i>admin_state</i>	The admin status; one of up, down

### Response

Response Body (JSON)	{ "vlan_name": "<vlan_name>", "vlan_id": "<vlan_id>", "admin_state": "<admin_state>", }
-------------------------	---

where:

Element	Description
<i>vlan_name</i>	The name of the VLAN.
<i>vlan_id</i>	VLAN number.; an integer from 2-3999.
<i>admin_state</i>	The admin status; one of up, down.

## Get VLAN

Get properties of a VLAN.

### Request

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

### Response

Response Body (JSON)	<pre>{   "vlan_name": "&lt;vlan_name&gt;",   "vlan_id": "&lt;vlan_id&gt;",   "admin_state": "&lt;admin_state&gt;",   "mst_inst_id": "&lt;mst_inst_id&gt;",   "interfaces": [     {       "if_name": "&lt;if_name&gt;",       "bridge_port_mode": "&lt;bridge_port_mode&gt;",       "pvid": "&lt;pvid&gt;"     }   ] }</pre>
-------------------------	---

## Update VLAN

Update 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
<i>vlan_name</i>	VLAN name; a string up to 32 characters long. To change a vlan name with default name, the <i>vlan_name</i> field must be null.
<i>admin_state</i>	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>", "bridge_port_mode": "<bridge_port_mode>", "pvid": "<pvid>" } ] }
-------------------------	--

## Delete VLAN

Delete a VLAN.

**Note:** If the specified *vlan\_id* is `a11`, all user-created VLANs will be deleted.

### *Request*

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

---

## VLAN Interface Properties

The following VLAN interface property URIs are available:

- `/nos/api/cfg/vlan_interface` GET, PUT

The following VLAN interface property commands are available:

- [Get VLAN Properties of All Interfaces](#)
- [Get VLAN Interface Properties](#)
- [Update VLAN Interface Properties](#)

## Get VLAN Properties of All Interfaces

Get 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)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "bridge_port_mode": "&lt;bridge_port_mode&gt;",     "pvid": "&lt;pvid&gt;",     "vlans": [       "&lt;vlan_id&gt;"     ]   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The Ethernet interface must exist.
<i>bridge_port_mode</i>	Bridge port mode; one of <b>access</b> , <b>trunk</b> .
<i>pvid</i>	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.
<i>vlans</i>	VLAN memberships; either <b>all</b> , <b>none</b> , or an integer from 1-3999.



## Get VLAN Interface Properties

Get 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)	<pre>{   "if_name": "&lt;if_name&gt;",   "bridge_port_mode": "&lt;bridge_port_mode&gt;",   "pvid": "&lt;pvid&gt;",   "vlans": ["&lt;vlan_id&gt;"] }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The Ethernet interface must exist.
<i>bridge_port_mode</i>	Bridge port mode; one of <code>access</code> , <code>trunk</code> .
<i>pvid</i>	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.
<i>vlans</i>	VLAN memberships; either <code>all</code> , <code>none</code> , or an integer from 1-3999.

## Update VLAN Interface Properties

Update 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>", "bridge_port_mode": "<bridge_port_mode>", "pvid": "<pvid>", "vlans": ["<vlan_id>"] }

where:

Element	Description
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The Ethernet interface must exist.
<i>bridge_port_mode</i>	Bridge port mode; one of access, trunk
<i>pvid</i>	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.
<i>vlans</i>	(Optional) VLAN memberships; all, 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>", "bridge_port_mode": "<bridge_port_mode>", "pvid": "<pvid>", "vlans": ["<vlans>"] }
-------------------------	--

---

## 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

Get 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)	<pre>{   "if_name": "&lt;if_name&gt;",   "edge_port": "&lt;edge_port&gt;",   "bpdu_guard": "&lt;bpdu_guard&gt;",   "loop_guard": "&lt;loop_guard&gt;",   "root_guard": "&lt;root_guard&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>edge_port</i>	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .
<i>bpdu_guard</i>	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>loop_guard</i>	(Optional) Whether loop guard is enabled on a port for additional checks for preventing STP looping; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>root_guard</i>	(Optional) Whether guard mode is set to root guard on interface.

## Get STP Interface Properties

Get 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)	<pre>{   "if_name": "&lt;if_name&gt;",   "edge_port": "&lt;edge_port&gt;",   "bpdu_guard": "&lt;bpdu_guard&gt;",   "loop_guard": "&lt;loop_guard&gt;",   "root_guard": "&lt;root_guard&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>edge_port</i>	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .
<i>bpdu_guard</i>	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>loop_guard</i>	(Optional) Whether loop guard is enabled on a port for additional checks for preventing STP looping; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>root_guard</i>	(Optional) Whether guard mode is set to root guard on interface.

## Update STP Interface Properties

Update 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
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>edge_port</i>	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .
<i>bpdu_guard</i>	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>loop_guard</i>	(Optional) Whether loop guard is enabled on a port for additional checks for preventing STP looping; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>root_guard</i>	(Optional) Whether guard mode is set to root guard on interface.

## Response

Response Body (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "edge_port": "&lt;edge_port&gt;",   "bpdu_guard": "&lt;bpdu_guard&gt;",   "loop_guard": "&lt;loop_guard&gt;",   "root_guard": "&lt;root_guard&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>edge_port</i>	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .
<i>bpdu_guard</i>	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>loop_guard</i>	(Optional) Whether loop guard is enabled on a port for additional checks for preventing STP looping; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>root_guard</i>	(Optional) Whether guard mode is set to root guard on interface.

---

## 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

Update global MSTP properties of the system.

### Request

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

### Response

Response Body (JSON)	<pre>{   "region_name": "&lt;region_name&gt;"   "revision": "&lt;revision&gt;" }</pre>
-------------------------	--

where:

Element	Description
<i>region_name</i>	Region name; a string up to 32 characters long.
<i>revision</i>	Revision number; an integer from 0-65535.

## Update MSTP System Properties

Update 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
<i>region_name</i>	Region name; a string up to 32 characters long.
<i>revision</i>	Revision number; an integer from 0-65535.

### Response

Response Body (JSON)	{ "region_name": "<region_name>" "revision": "<revision>" }
-------------------------	--

## Get Properties of All MSTP Instances

Get properties of all MSTP instances.

### Request

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

### Response

Response Body (JSON)	<pre>[   {     "instance_id": "&lt;instance_id&gt;",     "instance_prio": "&lt;instance_prio&gt;",     "vlans": [       {         "vlan_id": "&lt;vlan_id&gt;"       }     ]   } ]</pre>
-------------------------	--

where:

Element	Description
<i>instance_id</i>	MST instance ID; an integer from 0-64. Instance 0 refers to the CIST.
<i>instance_prio</i>	Sets the instance bridge priority; an integer from 0-61440. Default value: 32768.
<i>vlans</i>	Maps a range of VLANs to a multiple spanning tree instance (MSTI); an integer from 1-4094.

## Create MSTP Instance

Create an MSTP instance.

### Request

Method Type	POST
Request URI	/nos/api/cfg/mstp_instance
Request Body (JSON)	<pre>{   "instance_id": "&lt;instance_id&gt;",   "instance_prio": "&lt;instance_prio&gt;",   "vlans": [     {       "vlan_id": "&lt;vlan_id&gt;"     }   ] }</pre>

where:

Element	Description
<i>instance_id</i>	MST instance ID; an integer from 0-64. Instance 0 refers to the CIST.
<i>instance_prio</i>	Sets the instance bridge priority; an integer from 0-61440. Default value: 32768.
<i>vlans</i>	Maps a range of VLANs to a multiple spanning tree instance (MSTI); an integer from 1-4094.

### Response

Response Body (JSON)	<pre>{   "instance_id": "&lt;instance_id&gt;",   "instance_prio": "&lt;instance_prio&gt;",   "vlans": [     {       "vlan_id": "&lt;vlan_id&gt;"     }   ] }</pre>
-------------------------	--

## Get MSTP Instance

Get 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)	<pre>{   "instance_id": "&lt;instance_id&gt;",   "instance_prio": "&lt;instance_prio&gt;",   "vlans": [     {       "vlan_id": "&lt;vlan_id&gt;"     }   ] }</pre>
-------------------------	--

## Update MSTP Instance

Update 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)	<pre>{   "instance_id": "&lt;instance_id&gt;",   "instance_prio": "&lt;instance_prio&gt;",   "vlans": [     {       "vlan_id": "&lt;vlan_id&gt;"     }   ] }</pre>

where:

Element	Description
<i>instance_id</i>	MST instance ID; an integer from 0-64. Instance 0 refers to the CIST.
<i>instance_prio</i>	Sets the instance bridge priority; an integer from 0-61440. Default value: 32768.
<i>vlans</i>	Maps a range of VLANs to a multiple spanning tree instance (MSTI); an integer from 1-4094.

### Response

Response Body (JSON)	<pre>{   "instance_id": "&lt;instance_id&gt;",   "instance_prio": "&lt;instance_prio&gt;",   "vlans": [     {       "vlan_id": "&lt;vlan_id&gt;"     }   ] }</pre>
-------------------------	--

## Delete MSTP Instance

Delete an MSTP instance.

### *Request*

Method Type	DELETE
Request URI	/nos/api/cfg/mstp_instance/<instance_id>
Request Body (JSON)	

where:

Element	Description
<i>instance_id</i>	MST instance ID; an integer from 0-64. Instance 0 refers to the CIST.

## Get Interface Properties of an MSTP Instance

Get 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
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>path_cost</i>	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.
<i>port_prio</i>	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

Update 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
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>path_cost</i>	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.
<i>port_prio</i>	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>" }
-------------------------	--

---

## 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](#)
- [Update ARP System Properties](#)

## Get LLDP System Properties

Get 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
<i>reinit delay</i>	The number of seconds until LLDP re-initialization is attempted on an interface; an integer from 1-10. Default value: 2 seconds.
<i>transmit interval</i>	The time interval, in seconds, between transmissions of LLDP messages; an integer from 5-32768. Default value: 30 seconds.
<i>transmit delay</i>	The number of seconds for transmission delay; an integer from 1-8192. Default value: 2 seconds.

## Update LLDP System Properties

Update 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
<i>reinit delay</i>	The number of seconds until LLDP re-initialization is attempted on an interface; an integer from 1-10. Default value: 2 seconds.
<i>transmit interval</i>	The time interval, in seconds, between transmissions of LLDP messages; an integer from 5-32768.. Default value: 30 seconds.
<i>transmit delay</i>	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

Get 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
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The Ethernet interface must exist.
<i>ena_lldp_rx</i>	Enables or disables LLDP frame reception on a physical interface; one of <b>yes</b> (default), <b>no</b> .
<i>ena_lldp_tx</i>	Enables or disable sLLDP frame transmission on a physical interface; one of <b>yes</b> (default), <b>no</b> .

## Get LLDP Interface Properties

Get 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)	<pre>{   "if_name": "&lt;if_name&gt;",   "ena_lldp_rx": "&lt;ena_lldp_rx&gt;",   "ena_lldp_tx": "&lt;ena_lldp_tx&gt;" }</pre>
-------------------------	---

## Update LLDP Interface Properties

Update 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
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The Ethernet interface must exist.
<i>ena_lldp_rx</i>	Enables or disables LLDP frame reception on a physical interface; one of <b>yes</b> (default), <b>no</b> .
<i>ena_lldp_tx</i>	Enables or disable sLLDP frame transmission on a physical interface; one of <b>yes</b> (default), <b>no</b> .

### Response

Response Body (JSON)	{ "if_name": "<if_name>", "ena_lldp_rx": "<ena_lldp_rx>", "ena_lldp_tx": "<ena_lldp_tx>" }
-------------------------	--

## Get LLDP Interface Statistics

Get 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)	<pre>{   "total frames": "&lt;total_frames&gt;",   "total tlvs discarded": "&lt;total_tlvs_discarded&gt;",   "total frames transmitted": "&lt;total_frames_transmitted&gt;",   "total errored frames": "&lt;total_errored_frames&gt;",   "total frames discarded": "&lt;total_frames_discarded&gt;",   "total entries aged": "&lt;total_entries_aged&gt;",   "total tlvs unrecognized": "&lt;total_tlvs_unrecognized&gt;" }</pre>
-------------------------	---

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 errored 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

Get 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>" }
-------------------------	--

where:

Element	Description
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The Ethernet interface must exist.
<i>capability</i>	Remote switch capability; one of (B) - Bridge, (R) - Router.
<i>rx ttl</i>	The TTL.
<i>system name</i>	Remote system name.
<i>system description</i>	Remote system description.

## Get LLDP Neighbor Information for All Interfaces

Get 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)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "capability": "&lt;capability&gt;",     "rx ttl": "&lt;rx ttl&gt;",     "system name": "&lt;system name&gt;",     "system description": "&lt;system description&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Ethernet interface name (Str). <b>Note:</b> The Ethernet interface must exist.
<i>capability</i>	Remote switch capability; one of (B) - Bridge, (R) - Router.
<i>rx ttl</i>	The TTL.
<i>system name</i>	Remote system name.
<i>system description</i>	Remote system description.

---

## VRF

The following VRF URIs are available:

- /nos/api/cfg/vrf GET
- /nos/api/cfg/vrf/<vrf\_name> GET

The following VRF commands are available:

- [Get All VRFs](#)
- [Get VRF](#)

## Get All VRFs

Get properties of all VRFs.

### *Request*

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

### *Response*

Response Body (JSON)	<pre>[   {     "vrf_name": "&lt;vrf_name&gt;",     "interfaces": ["&lt;if_name&gt;"]   } ]</pre>
-------------------------	--

where:

Element	Description
<i>vrf_name</i>	VRF name; a string up to 64 characters long.
<i>interfaces</i>	Interface members of the VRF. <b>Note:</b> The interfaces must exist.

## Get VRF

Get properties of one VRF.

### *Request*

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

### *Response*

Response Body (JSON)	<pre>{   "vrf_name": "&lt;vrf_name&gt;",   "interfaces": [     "&lt;if_name&gt;"   ] }</pre>
-------------------------	--

---

## 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

Get IP properties of all interfaces.

### Request

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

### Response

Response Body (JSON)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "bridge_port": "&lt;bridge_port&gt;",     "mtu": {mtu},     "ip_addr": "&lt;ip_addr&gt;",     "ip_prefix_len": "&lt;ip_prefix_len&gt;",     "vrf_name": "&lt;vrf_name&gt;",     "admin_state": "&lt;admin_state&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	IP interface name (Str). <b>Note:</b> The interface must exist.
<i>bridge_port</i>	Whether or not the port is a bridge port; one of <b>yes</b> (default), <b>no</b> .
<i>mtu</i>	The maximum transmission unit, in bytes; an integer from 64-9216. Default value: 1500.
<i>ip_addr</i>	IP address for the interface.
<i>ip_prefix_len</i>	IP address mask; a positive integer from 1-32.
<i>vrf_name</i>	The name of the VRF to which the interface belongs. <b>Note:</b> The named VRF must exist.
<i>admin_state</i>	The admin status; one of <b>up</b> , <b>down</b> .

## Get IP Interface Properties

Get 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>{   "if_name": "&lt;if_name&gt;",   "bridge_port": "&lt;bridge_port&gt;",   "mtu": &lt;mtu&gt;,   "ip_addr": "&lt;ip_addr&gt;",   "ip_prefix_len": "&lt;ip_prefix_len&gt;",   "vrf_name": "&lt;vrf_name&gt;",   "admin_state": "&lt;admin_state&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	IP interface name (Str). <b>Note:</b> The interface must exist.
<i>bridge_port</i>	Whether or not the port is a bridge port; one of yes (default), no.
<i>mtu</i>	The maximum transmission unit, in bytes; an integer from 64-9216. Default value: 1500.
<i>ip_addr</i>	IP address for the interface.
<i>ip_prefix_len</i>	IP address mask; a positive integer from 1-32.
<i>vrf_name</i>	The name of the VRF to which the interface belongs. <b>Note:</b> The named VRF must exist.
<i>admin_state</i>	The admin status; one of up, down.



## Update IP Interface Properties

Update 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>{   "if_name": "&lt;if_name&gt;",   "bridge_port": "&lt;bridge_port&gt;",   "mtu": &lt;mtu&gt;,   "ip_addr": "&lt;ip_addr&gt;",   "ip_prefix_len": "&lt;ip_prefix_len&gt;",   "vrf_name": "&lt;vrf_name&gt;",   "admin_state": "&lt;admin_state&gt;" }</pre>

where:

Element	Description
<i>if_name</i>	IP interface name (Str). <b>Note:</b> The interface must exist.
<i>bridge_port</i>	Whether or not the port is a bridge port; one of yes (default), no.
<i>mtu</i>	The maximum transmission unit, in bytes; an integer from 64-9216. Default value: 1500.
<i>ip_addr</i>	IP address for the interface.
<i>ip_prefix_len</i>	IP address mask; a positive integer from 1-32.
<i>vrf_name</i>	The name of the VRF to which the interface belongs. <b>Note:</b> The named VRF must exist.
<i>admin_state</i>	The admin status; one of up, down.

### Response

Response Body (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "bridge_port": "&lt;bridge_port&gt;",   "mtu": &lt;mtu&gt;,   "ip_addr": "&lt;ip_addr&gt;",   "ip_prefix_len": "&lt;ip_prefix_len&gt;",   "vrf_name": "&lt;vrf_name&gt;",   "admin_state": "&lt;admin_state&gt;" }</pre>
-------------------------	---

---

## 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

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 ARP System Properties

Get 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
<i>ageout_time</i>	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

## Update ARP System Properties

Update 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
<i>ageout_time</i>	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

Get 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
<i>if_name</i>	IP interface name (Str). <b>Note:</b> The interface must exist.
<i>ageout_time</i>	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

## Get ARP Interface Properties

Get 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
<i>if_name</i>	IP interface name (Str). <b>Note:</b> The interface must exist.
<i>ageout_time</i>	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

## Update ARP Interface Properties

Update 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
<i>if_name</i>	The IP interface name (Str). <b>Note:</b> The interface must exist.
<i>ageout_time</i>	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
<i>if_name</i>	The IP interface name (Str).
<i>ageout_time</i>	The global ARP entry age-out time, in seconds; an integer from 60-28800. Default value: 1500 seconds.

---

## Static ARP

The following static ARP URIs are available:

- `/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

The following static ARP commands are available:

- [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 Static ARP Entries of All Interfaces

Get 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)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "ip_addr": "&lt;ip_addr&gt;",     "mac_addr": "&lt;mac_addr&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>ip_addr</i>	The IP address.
<i>mac_addr</i>	The MAC address (in <i>xxxx,xxx,xxx</i> format).

## Get Static ARP Entries of One Interface

Get 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)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "ip_addr": "&lt;ip_addr&gt;",     "mac_addr": "&lt;mac_addr&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>ip_addr</i>	The IP address.
<i>mac_addr</i>	The MAC address (in <i>xxxx_xxxx_xxxx</i> format).

## Create Static ARP Entry

Create 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
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>ip_addr</i>	The IP address.
<i>mac_addr</i>	The MAC address (in <i>xxxx.xxxx.xxxx</i> format).

### Response

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

## Get Static ARP Entry

Get 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
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>ip_addr</i>	The IP address.
<i>mac_addr</i>	The MAC address (in <i>xxxx_xxxx_xxxx</i> format).

## Update Static ARP Entry

Update 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
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>ip_addr</i>	The IP address.
<i>mac_addr</i>	The MAC address (in <i>xxxx.xxxx.xxxx</i> format).

### Response

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

## Delete Static ARP Entry

Delete 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)	

---

## 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

Get 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)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "vr_id": "&lt;vr_id&gt;",     "ip_addr": "&lt;ip_addr&gt;",     "ad_intvl": "&lt;ad_intvl&gt;",     "preempt": "&lt;preempt&gt;",     "prio": "&lt;prio&gt;",     "admin_state": "&lt;admin_state&gt;",     "oper_state": "&lt;oper_state&gt;",     "track_if": "&lt;track_if&gt;",     "accept_mode": "&lt;accept_mode&gt;",     "switch_back_delay": "&lt;switch_back_delay&gt;",     "v2_compt": "&lt;v2_compt&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>vr_id</i>	The VRRP session Virtual Router (VR) ID; an integer from 1-255. Default value is 0.
<i>ip_addr</i>	The IP address of the VR; a valid IPv4 address.
<i>ad_intvl</i>	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from 5-4095. Default value: 100 centi-seconds.
<i>preempt</i>	Enable the preemption of a lower priority master; one of <b>yes</b> (default) , <b>no</b> .
<i>prio</i>	The priority of the VR on the switch; an integer from 1-254. Default value: 100.
<i>admin_state</i>	Enable the VR one of <b>up</b> (default), <b>down</b> .



<b>Element</b>	<b>Description</b>
<i>oper_state</i>	The operation state of the VR; one of <b>master</b> , <b>backup</b> , <b>init</b> .
<i>track_if</i>	The interface to track by this VR. Default value: <b>none</b> . <b>Note:</b> If an interface is specified, it must exist.
<i>accept_mode</i>	Enables or disables the accept mode for this session; one of <b>yes</b> (default), <b>no</b> .
<i>switch_back_delay</i>	The switch back delay interval; an integer from 1-500000, or 0 to disable (default).
<i>v2_compt</i>	Enables backward compatibility for VRRPv2 for the VR; one of <b>yes</b> , <b>no</b> (default).

## Get VRRP VRs of One Interface

Get 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
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.

### Response

Response Body (JSON)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "vr_id": "&lt;vr_id&gt;",     "ip_addr": "&lt;ip_addr&gt;",     "ad_intvl": "&lt;ad_intvl&gt;",     "preempt": "&lt;preempt&gt;",     "prio": "&lt;prio&gt;",     "admin_state": "&lt;admin_state&gt;",     "oper_state": "&lt;oper_state&gt;",     "track_if": "&lt;track_if&gt;",     "accept_mode": "&lt;accept_mode&gt;",     "switch_back_delay": "&lt;switch_back_delay&gt;",     "v2_compt": "&lt;v2_compt&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>vr_id</i>	Virtual Router (VR) identifier; an integer from 1-255.
<i>ip_addr</i>	The IP address of the VR; a valid IPv4 address.
<i>ad_intvl</i>	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from 5-4095. Default value: 100 centi-seconds.

<b>Element</b>	<b>Description</b>
<i>preempt</i>	Enable the preemption of a lower priority master; one of <b>yes</b> (default) , <b>no</b> .
<i>prio</i>	The priority of the VR on the switch; an integer from 1-254. Default value: 100.
<i>admin_state</i>	Enable the VR one of <b>up</b> (default), <b>down</b> .
<i>oper_state</i>	The operation state of the VR; one of <b>master</b> , <b>backup</b> , <b>init</b> .
<i>track_if</i>	The interface to track by this VR. Default value: <b>none</b> . <b>Note:</b> If an interface is specified, it must exist.
<i>accept_mode</i>	Enables or disables the accept mode for this session; one of <b>yes</b> (default), <b>no</b> .
<i>switch_back_delay</i>	The switch back delay interval; an integer from 1-500000, or 0 to disable (default).
<i>v2_compt</i>	Enables backward compatibility for VRRPv2 for the VR; one of <b>yes</b> , <b>no</b> (default).

## Create VRRP VR

Create a VRRP VR.

### Request

Method Type	POST
Request URI	/nos/api/cfg/vrrp/<if_name>
Request Body (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "vr_id": "&lt;vr_id&gt;",   "ip_addr": "&lt;ip_addr&gt;",   "ad_intvl": "&lt;ad_intvl&gt;",   "preempt": "&lt;preempt&gt;",   "prio": "&lt;prio&gt;",   "admin_state": "&lt;admin_state&gt;",   "track_if": "&lt;track_if&gt;",   "accept_mode": "&lt;accept_mode&gt;",   "switch_back_delay": "&lt;switch_back_delay&gt;",   "v2_compt": "&lt;v2_compt&gt;" }</pre>

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>vr_id</i>	Virtual Router (VR) identifier; an integer from 1-255.
<i>ip_addr</i>	The IP address of the VR; a valid IPv4 address.
<i>ad_intvl</i>	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from 5-4095. Default value: 100 centi-seconds.
<i>preempt</i>	Enable the preemption of a lower priority master; one of <b>yes</b> (default) , <b>no</b> .
<i>prio</i>	The priority of the VR on the switch; an integer from 1-254. Default value: 100.
<i>admin_state</i>	Enable the VR one of <b>up</b> (default), <b>down</b> .
<i>oper_state</i>	The operation state of the VR; one of <b>master</b> , <b>backup</b> , <b>init</b> .
<i>track_if</i>	The interface to track by this VR. Default value: <b>none</b> . <b>Note:</b> If an interface is specified, it must exist.
<i>accept_mode</i>	Enables or disables the accept mode for this session; one of <b>yes</b> (default), <b>no</b> .

Element	Description
<i>switch_back_delay</i>	The switch back delay interval; an integer from 1-500000, or 0 to disable (default).
<i>v2_compt</i>	Enables backward compatibility for VRRPv2 for the VR; one of yes, no (default).

## Response

Response Body (JSON)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "vr_id": "&lt;vr_id&gt;",     "ip_addr": "&lt;ip_addr&gt;",     "ad_intvl": "&lt;ad_intvl&gt;",     "preempt": "&lt;preempt&gt;",     "prio": "&lt;prio&gt;",     "admin_state": "&lt;admin_state&gt;",     "oper_state": "&lt;oper_state&gt;",     "track_if": "&lt;track_if&gt;",     "accept_mode": "&lt;accept_mode&gt;",     "switch_back_delay": "&lt;switch_back_delay&gt;",     "v2_compt": "&lt;v2_compt&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>vr_id</i>	Virtual Router (VR) identifier; an integer from 1-255.
<i>ip_addr</i>	The IP address of the VR; a valid IPv4 address.
<i>ad_intvl</i>	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from 5-4095. Default value: 100 centi-seconds.
<i>preempt</i>	Enable the preemption of a lower priority master; one of yes (default) , no.
<i>prio</i>	The priority of the VR on the switch; an integer from 1-254. Default value: 100.
<i>admin_state</i>	Enable the VR one of up (default), down.
<i>oper_state</i>	The operation state of the VR; one of master, backup, init.
<i>track_if</i>	The interface to track by this VR. Default value: none. <b>Note:</b> If an interface is specified, it must exist.
<i>accept_mode</i>	Enables or disables the accept mode for this session; one of yes (default), no.

<b>Element</b>	<b>Description</b>
<i>switch_back_delay</i>	The switch back delay interval; an integer from 1-500000, or 0 to disable (default).
<i>v2_compt</i>	Enables backward compatibility for VRRPv2 for the VR; one of yes, no (default).

## Get VRRP VR

Get 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)	<pre>{   "if_name": "&lt;if_name&gt;",   "vr_id": "&lt;vr_id&gt;",   "ip_addr": "&lt;ip_addr&gt;",   "ad_intvl": "&lt;ad_intvl&gt;",   "preempt": "&lt;preempt&gt;",   "prio": "&lt;prio&gt;",   "admin_state": "&lt;admin_state&gt;",   "oper_state": "&lt;oper_state&gt;",   "track_if": "&lt;track_if&gt;",   "accept_mode": "&lt;accept_mode&gt;",   "switch_back_delay": "&lt;switch_back_delay&gt;",   "v2_compt": "&lt;v2_compt&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>vr_id</i>	Virtual Router (VR) identifier; an integer from 1-255.
<i>ip_addr</i>	The IP address of the VR; a valid IPv4 address.
<i>ad_intvl</i>	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from 5-4095. Default value: 100 centi-seconds.
<i>preempt</i>	Enable the preemption of a lower priority master; one of <b>yes</b> (default) , <b>no</b> .
<i>prio</i>	The priority of the VR on the switch; an integer from 1-254. Default value: 100.
<i>admin_state</i>	Enable the VR one of <b>up</b> (default), <b>down</b> .
<i>oper_state</i>	The operation state of the VR; one of <b>master</b> , <b>backup</b> , <b>init</b> .

<b>Element</b>	<b>Description</b>
<i>track_if</i>	The interface to track by this VR. Default value: none. <b>Note:</b> If an interface is specified, it must exist.
<i>accept_mode</i>	Enables or disables the accept mode for this session; one of yes (default), no.
<i>switch_back_delay</i>	The switch back delay interval; an integer from 1-500000, or 0 to disable (default).
<i>v2_compt</i>	Enables backward compatibility for VRRPv2 for the VR; one of yes, no (default).



## Update VRRP VR

Update the properties of a VRRP VR.

### Request

Method Type	PUT
Request URI	/nos/api/cfg/vrrp/<if_name>/<vrid>
Request Body (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "vr_id": "&lt;vr_id&gt;",   "ip_addr": "&lt;ip_addr&gt;",   "ad_intvl": "&lt;ad_intvl&gt;",   "preempt": "&lt;preempt&gt;",   "prio": "&lt;prio&gt;",   "admin_state": "&lt;admin_state&gt;",   "track_if": "&lt;track_if&gt;",   "accept_mode": "&lt;accept_mode&gt;",   "switch_back_delay": "&lt;switch_back_delay&gt;",   "v2_compt": "&lt;v2_compt&gt;" }</pre>

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>vr_id</i>	Virtual Router (VR) identifier; an integer from 1-255.
<i>ip_addr</i>	The IP address of the VR; a valid IPv4 address.
<i>ad_intvl</i>	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from 5-4095. Default value: 100 centi-seconds.
<i>preempt</i>	Enable the preemption of a lower priority master; one of <b>yes</b> (default), <b>no</b> .
<i>prio</i>	The priority of the VR on the switch; an integer from 1-254. Default value: 100.
<i>admin_state</i>	Enable the VR; one of <b>up</b> (default), <b>down</b> .
<i>oper_state</i>	The operation state of the VR; one of <b>master</b> , <b>backup</b> , <b>init</b> .
<i>track_if</i>	The interface to track by this VR. Default value: none. <b>Note:</b> If an interface is specified, it must exist.
<i>accept_mode</i>	Enables or disables the accept mode for this session; one of <b>yes</b> (default), <b>no</b> .

Element	Description
<i>switch_back_delay</i>	The switch back delay interval; an integer from 1-500000, or 0 to disable (default).
<i>v2_compt</i>	Enables backward compatibility for VRRPv2 for the VR; one of yes, no (default).

## Response

Response Body (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "vr_id": "&lt;vr_id&gt;",   "ip_addr": "&lt;ip_addr&gt;",   "ad_intvl": "&lt;ad_intvl&gt;",   "preempt": "&lt;preempt&gt;",   "prio": "&lt;prio&gt;",   "admin_state": "&lt;admin_state&gt;",   "oper_state": "&lt;oper_state&gt;",   "track_if": "&lt;track_if&gt;",   "accept_mode": "&lt;accept_mode&gt;",   "switch_back_delay": "&lt;switch_back_delay&gt;",   "v2_compt": "&lt;v2_compt&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>vr_id</i>	Virtual Router (VR) identifier; an integer from 1-255.
<i>ip_addr</i>	The IP address of the VR; a valid IPv4 address.
<i>ad_intvl</i>	Advertisement interval (The number of centi-seconds between advertisements for VRRPv3); a multiple of 5 from 5-4095. Default value: 100 centi-seconds.
<i>preempt</i>	Enable the preemption of a lower priority master; one of yes (default) , no.
<i>prio</i>	The priority of the VR on the switch; an integer from 1-254. Default value: 100.
<i>admin_state</i>	Enable the VR one of up (default), down.
<i>oper_state</i>	The operation state of the VR; one of master, backup, init.
<i>track_if</i>	The interface to track by this VR. Default value: none. <b>Note:</b> If an interface is specified, it must exist.
<i>accept_mode</i>	Enables or disables the accept mode for this session; one of yes (default), no.

<b>Element</b>	<b>Description</b>
<i>switch_back_delay</i>	The switch back delay interval; an integer from 1-500000, or 0 to disable (default).
<i>v2_compt</i>	Enables backward compatibility for VRRPv2 for the VR; one of yes, no (default).

## Delete VRRP VR

Delete 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)	

---

## 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

Get 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
<i>vid</i>	VLAN number; an integer from 1-3999.
<i>if_name</i>	Ethernet interface name or port aggregation name.

## Response

Response Body (JSON)	<pre>[   {     "vid": "&lt;vid&gt;",     "if_name": "&lt;if_name&gt;",     "group_address": "&lt;group_address&gt;",     "source_ip":       {         "include_list" :           [             {               "source_address" : "&lt;source ip address&gt;",               "uptime" : "&lt;uptime&gt;",               "expires" : "&lt;expires&gt;",               "fwd" : "&lt;fwd&gt;",               "flags" : "&lt;flags&gt;"             }           ],         "exclude_list" :           [             {               "source_address" : "&lt;source ip address&gt;",               "uptime" : "&lt;uptime&gt;",               "expires" : "&lt;expires&gt;",               "fwd" : "&lt;fwd&gt;",               "flags" : "&lt;flags&gt;"             }           ]       }     "flags" : "&lt;flags&gt;",     "expires": "&lt;expires&gt;",     "version": "&lt;version&gt;",     "filter_mode": "&lt;filter_mode&gt;",   } ]</pre>
-------------------------	---

where:

Element	Description
<i>vid</i>	VLAN number; an integer from 1-3999.
<i>if_name</i>	Ethernet interface name or port aggregation name.
<i>group_address</i>	The IGMP group IPv4 address.
<i>source_ip</i>	Dictionary of included and excluded source IP details.
<i>include_list</i>	List of included source IP details.
<i>exclude_list</i>	List of excluded source IP details.
<i>source_address</i>	Included or excluded source IPv4 address.
<i>uptime</i>	Time since switch is running in HH:MM:SS format.
<i>expires</i>	Source expiry time interval in HH:MM:SS format.
<i>fwd</i>	Whether to forward traffic for this source IP; one of yes, no.

<b>Element</b>	<b>Description</b>
<i>flags</i> (in <i>include_list</i> and <i>exclude_list</i> )	Source IP flag; one of: <ul style="list-style-type: none"> <li>● D – Dynamic</li> <li>● S – Static</li> <li>● L – Local and Static</li> </ul>
<i>flags</i>	Group flags; one of <i>Dynamic</i> , <i>Static</i> IGMP group.
<i>expires</i>	Group expiry time interval HH:MM:SS format.
<i>version</i>	IGMP version number.
<i>filter_mode</i>	IGMP Router-Filter-Mode State; one of <i>include</i> , <i>exclude</i> .



## Get IGMP Mrouter

Get 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
<i>vlan_id</i>	VLAN number; an integer from 1-3999.
<i>if_name</i>	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
<i>vid</i>	VLAN number; an integer from 1-3999.
<i>if_name</i>	Ethernet interface name or port aggregation name.
<i>mrouter_address</i>	The IGMP multicast router IPv4 address.
<i>mrouter_type</i>	Specifies how this entry was learned; one of dynamic, PIM hello, static.
<i>expires</i>	Expiry time interval HH:MM:SS format.

## Update IGMP Mrouter Interface for a VLAN

Add 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
<i>vid</i>	VLAN number; an integer from 1-3999.
<i>if_name</i>	Ethernet interface name or port aggregation name.

### Response

Response Body (JSON)	{ "if_name": "<if_name>", }
-------------------------	-----------------------------------

where:

Element	Description
<i>if_name</i>	Ethernet interface name or port aggregation name.

## Get IGMP Querier

Get 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
<i>vlan_id</i>	VLAN number; an integer from 1-3999.

### Response

Response Body (JSON)	<pre>[   {     "vid": "&lt;vid&gt;",     "address": "&lt;address&gt;",     "state": "&lt;state&gt;",     "version": "&lt;version&gt;",     "expires": "&lt;expires&gt;",   } ]</pre>
-------------------------	--

where:

Element	Description
<i>vid</i>	VLAN number; an integer from 1-3999.
<i>address</i>	Querier IPv4 address.
<i>state</i>	Elected querier state; one of <b>Querier</b> , <b>Non-querier</b> .
<i>version</i>	Snooping querier version.
<i>expires</i>	Expiry time interval HH:MM:SS format.

## 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
<i>address</i>	The querier IPv4 address.

### Response

Response Body (JSON)	[ { "vid": "<vid>", "address": "<address>", "state": "<state>", "version": "<version>", "expires": "<expires>", } ]
-------------------------	---

where:

Element	Description
<i>vid</i>	VLAN number; an integer from 1-3999.
<i>address</i>	Querier IPv4 address.
<i>state</i>	Elected querier state; one of <b>Querier</b> , <b>Non-querier</b> .
<i>version</i>	Snooping querier version.
<i>expires</i>	Expiry time interval HH:MM:SS format.

## Get IGMP Snooping System Properties

Get 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
<i>ena_igmp_snoop</i>	Enables IGMP snooping globally on all VLANs; one of <i>yes</i> (default), <i>no</i> .  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

Update 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
<i>ena_igmp_snoop</i>	Enables IGMP snooping globally on all VLANs; one of <i>yes</i> (default), <i>no</i> .  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.

### Response

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

## Get IGMP Snooping Properties of All VLANs

Get 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)	<pre>[   {     "vlan_id": "&lt;vlan_id&gt;",     "ena_igmp_snoop": "&lt;ena_igmp_snoop&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>vlan_id</i>	VLAN number.
<i>ena_igmp_snoop</i>	Enables IGMP snooping on a VLAN; one of yes (default), no.

## Get IGMP Snooping VLAN Properties

Get 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
<i>vlan_id</i>	VLAN number.
<i>ena_igmp_snoop</i>	Enables IGMP snooping on a VLAN; one of yes (default), no.



## Update IGMP Snooping VLAN Properties

Update 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
<i>vlan_id</i>	VLAN number. <b>Note:</b> The VLAN must exist.
<i>ena_igmp_snoop</i>	(Optional) Whether to enable IGMP snooping on a VLAN; one of yes, no. Default value: yes.
<i>fast_leave</i>	One of yes, no. Default value: no.
<i>query_interval</i>	(Optional) IGMP query interval, in seconds; an integer from 1-18000. Default value: 125.
<i>version</i>	(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
<i>vlan_id</i>	VLAN number. <b>Note:</b> The VLAN must exist.
<i>ena_igmp_snoop</i>	(Optional) Whether to enable IGMP snooping on a VLAN; one of yes, no. Default value: yes.

<b>Element</b>	<b>Description</b>
<i>fast_leave</i>	One of <b>yes</b> , <b>no</b> . Default value: <b>no</b> .
<i>query_interval</i>	(Optional) IGMP query interval, in seconds; an integer from 1-18000. Default value: 125.
<i>version</i>	(Optional) IGMP Snooping version number; one of 2, 3. Default value: 3.

---

## 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](#)
- [Get All VLANs](#)
- [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

Get 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)	<pre>{   "user_name": "&lt;user_name&gt;",   "group_name": "&lt;group_name&gt;",   "auth_type": "&lt;auth_type&gt;",   "auth_passwd": "&lt;password&gt;",   "priv_type": " &lt;priv_type&gt;",   "priv_passwd": "&lt;password&gt;"   "xclarity_id": "&lt;identifier&gt;" }</pre>
-------------------------	--

where:

Element	Description
<i>user_name</i>	User name; a text string from 5-32 characters long.
<i>group_name</i>	Group name; one of "network_operator", "network_admin". Default value: network-operator.
<i>auth_type</i>	Authentication type; one of "invalid", "md5", "sha".
<i>auth_passwd</i>	Authentication password; a string from 8-32 characters long.
<i>priv_type</i>	Privilege type; one of "invalid", "des", "aes".
<i>priv_passwd</i>	Privilege password; a string from 8-32 characters long.
<i>xclarity_id</i>	XClarity instance identifier; a 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

Set 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)	<pre>{   "user_name": "&lt;user_name&gt;",   "group_name": "&lt;group_name&gt;",   "auth_type": "&lt;auth_type&gt;",   "auth_passwd": "&lt;password&gt;",   "priv_type": " &lt;priv_type&gt;",   "priv_passwd": "&lt;password&gt;"   "xclarity_id": "&lt;identifier&gt;" }</pre>

where:

Element	Description
<i>user_name</i>	User name; a text string from 5-32 characters long.
<i>group_name</i>	Group name; one of "network_operator," "network_admin". Default value: network-operator.
<i>auth_type</i>	Authentication type; one of "invalid", "md5", "sha".
<i>auth_passwd</i>	Authentication password; a string from 8-32 characters long.
<i>priv_type</i>	Privilege type; one of "invalid", "des", "aes".
<i>priv_passwd</i>	Privilege password; a string from 8-32 characters long.
<i>xclarity_id</i>	XClarity instance identifier; a text string from 4-32 characters long.

## Response

Response Body (JSON)	<pre>{   "username": "&lt;username&gt;",   "group_name": "&lt;group_name&gt;",   "auth_type": "&lt;auth_type&gt;",   "auth_passwd": "&lt;password&gt;",   "priv_type": " " "&lt;priv_type&gt;",   "priv_passwd": "&lt;password&gt;",   "xclarity_id": "&lt;identifier&gt;" }</pre>
----------------------	--

where:

Element	Description
<i>username</i>	User name; a text string from 5-32 characters long.
<i>group_name</i>	Group name; one of "network_operator," "network_admin". Default value: network-operator.
<i>auth_type</i>	Authentication type; one of "invalid", "md5", "sha".
<i>auth_passwd</i>	Encrypted text string of authentication password; a string from 8-32 characters long.
<i>priv_type</i>	Privilege type; one of "invalid", "des", "aes".
<i>priv_passwd</i>	Encrypted text string of privilege password; a string from 8-32 characters long.
<i>xclarity_id</i>	XClarity instance identifier; a text string from 4-32 characters long.

## Delete the SNMPv3 Account for XClarity

Delete 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

Get 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)	<pre>{   "host_name": "&lt;host_name&gt;",   "security_level": "&lt;security_level&gt;",   "username": "&lt;username&gt;",   "message_type": " &lt;priv_type&gt;",   "port": "&lt;port&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>host_name</i>	Trap host name; a valid IPv4 or IPv6 address.
<i>security_level</i>	Security level; one of "auth", "authpriv", "noauth".
<i>username</i>	Username; a text string 5-32 characters long.
<i>message_type</i>	Message type; one of "trap", "inform". Default value: trap.
<i>port</i>	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

Set 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
<i>host_name</i>	Trap host name; a valid IPv4 or IPv6 address.
<i>security_level</i>	Security level; one of "auth", "authpriv", "noauth".
<i>username</i>	Username; a text string 5-32 characters long.
<i>message_type</i>	(Optional) Message type; one of "trap", "inform". Default value: trap.
<i>port</i>	(Optional) 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
<i>host_name</i>	Trap host name; a valid IPv4 or IPv6 address.
<i>security_level</i>	Security level; one of "auth", "authpriv", "noauth".

<b>Element</b>	<b>Description</b>
<i>username</i>	Username; a text string 5-32 characters long.
<i>message_type</i>	Message type; one of "trap", "inform". Default value: trap.
<i>port</i>	Host UDP port; an integer from 1-65535. Default value: 162.

## Delete the SNMPv3 Trap Host IP Address for XClarity

Delete 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.

---

## 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/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
- /nos/api/cfg/bgp/neighbor/details GET

The following BGP interface property commands are available:

- [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 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](#)
- [Get BGP Neighbor Configuration](#)

## Get BGP Global Statistics

Get global BGP statistics.

### Request

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

where:

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

### Response

Response Body (JSON)	<pre>[   {     "vrf_name": "&lt;vrf_name&gt;",     "stats": {       "in_msgs": "&lt;in_msgs&gt;",       "out_msgs": "&lt;sent_msg&gt;",       "bytes_in": "&lt;bytes_in&gt;",       "bytes_out": "&lt;bytes_out&gt;",       "open_in": "&lt;open_in&gt;",       "open_out": "&lt;open_out&gt;",       "update_in": "&lt;update_in&gt;",       "update_out": "&lt;update_out&gt;",       "keepalive_in": "&lt;keepalive_in&gt;",       "keepalive_out": "&lt;keepalive_out&gt;",       "notify_in": "&lt;notify_in&gt;",       "notify_out": "&lt;notify_out&gt;",       "refresh_in": "&lt;refresh_in&gt;",       "refresh_out": "&lt;refresh_out&gt;",       "dynamic_cap_in": "&lt;dynamic_cap_in&gt;",       "dynamic_cap_out": "&lt;dynamic_cap_out&gt;",     }   } ]</pre>
-------------------------	---

where:

Element	Description
<i>vrf_name</i>	VRF name (str).
<i>in_msgs</i>	Received message number; a positive integer.
<i>out_msgs</i>	Send message number; a positive integer.

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

## Clear BGP Global Statistics

Clear global BGP statistics.

### *Request*

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

where:

Element	Description
<i>vrf_name</i>	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

Get 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
<i>neighbor_ip</i>	Neighbor IP address; a valid IPv4 or IPv6 address.
<i>vrf_name</i>	(Optional) Address family name; one of <code>ipv4</code> or <code>ipv6m</code> . Default value: <code>ipv4</code> .
<i>af_name</i>	(Optional) VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .
<i>subaf_name</i>	(Optional) Subaddress family name; one of <code>unicast</code> , <code>multicast</code> . Default value: <code>unicast</code> .

## Response

Response Body (JSON)	<pre>[   {     "routes":     [       {         "origin": "&lt;origin&gt;",         "network": "&lt;network&gt;",         "mask_len": "&lt;mask_len&gt;",         "weight": "&lt;attr_weight&gt;",         "Metric": "&lt;metric&gt;",         "nexthop": "&lt;nexthop&gt;",         "aspath4B": "&lt;aspath4B&gt;",         "status": "&lt;falg&gt;",         "local_pref": "&lt;local_pref&gt;",         "aspath": "&lt;aspath&gt;",       }     ]   } ]</pre>
-------------------------	---

where:

Element	Description
<i>origin</i>	Route origin attribute; one of: <ul style="list-style-type: none"> <li>● <b>i</b> - IGP</li> <li>● <b>e</b> - EGP</li> <li>● <b>?</b> - incomplete</li> </ul>
<i>network</i>	Route destination IP address; a valid IPv4 or IPv6 address.
<i>mask_len</i>	Route mask length; an integer from 0-32.
<i>weight</i>	Route weight attribute; an integer from 0-65535.
<i>metric</i>	Route Multi-Exit Discriminator attribute; an integer from 0~4294967295.
<i>nexthop</i>	Route next hop; a valid IP address.
<i>aspath4B</i>	Route 4B AS path; an AS path VTY string.
<i>status</i>	Router status; one of: <ul style="list-style-type: none"> <li>● <b>s</b> - suppressed</li> <li>● <b>d</b> - damped</li> <li>● <b>h</b> - history</li> <li>● <b>*</b> - valid</li> <li>● <b>&gt;</b> - best</li> <li>● <b>i</b> - internal</li> </ul>
<i>local_pref</i>	Route local preference attribute; an integer from 0-4294967295.
<i>aspath</i>	Route AS path attribute; an AS path VTY string.

## Get BGP Neighbor RIB Advertised Information

Get 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
<i>neighbor</i>	Neighbor IP address; a valid IPv4 or IPv6 address.
<i>af_name</i>	(Optional) Address family name; one of <code>ipv4</code> or <code>ipv6m</code> . Default value: <code>ipv4</code> .
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .
<i>subaf_name</i>	(Optional) Subaddress family name; one of <code>unicast</code> , <code>multicast</code> . Default value: <code>unicast</code> .

## Response

Response Body (JSON)	<pre>[   {     "routes":       [         {           "origin": "&lt;origin&gt;",           "network": "&lt;prefix_addr&gt;",           "mask_len": "&lt;mask_len&gt;",           "weight": "&lt;weight&gt;",           "Metric": "&lt;metric&gt;",           "nexthop": "&lt;nexthop&gt;",           "aspath4B": "&lt;aspath4B&gt;",           "status": "&lt;status&gt;",           "local_pref": "&lt;local_pref&gt;",           "aspath": "&lt;aspath&gt;",         }       ]     }   ]</pre>
-------------------------	--

where:

Element	Description
<i>origin</i>	Route origin attribute; one of: <ul style="list-style-type: none"> <li>● i - IGP</li> <li>● e - EGP</li> <li>● ? - incomplete</li> </ul>
<i>network</i>	Route destination IP address; a valid IPv4 or IPv6 address.
<i>mask_len</i>	Route mask length; an integer from 0-32.
<i>weight</i>	Route weight attribute; an integer from 0-65535.
<i>metric</i>	Route med attribute; an integer from 0-4294967295.
<i>nexthop</i>	Route nexthop address; a valid IPv4 or IPv6 address.
<i>aspath4B</i>	Route 4B AS path; an AS path VTY string.
<i>status</i>	Router status; one of: <ul style="list-style-type: none"> <li>● s - suppressed</li> <li>● d - damped</li> <li>● h - history</li> <li>● * - valid</li> <li>● &gt; - best</li> <li>● i - internal</li> </ul>
<i>local_pref</i>	Route local preference attribute; an integer from 0-4294967295.
<i>aspath</i>	Route AS path attribute; a valid AS path VTY string.

## Get BGP Global Configuration

Get the BGP global configuration.

### Request

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

where:

Element	Description
<i>vrf_name</i>	VRF name; one of the VRF name, default, all. Default value: default.

### Response

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

where:

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

<b>Element</b>	<b>Description</b>
<i>hold-down timer</i>	Hold time, in seconds; an integer from 0-3600.
<i>as-local-count</i>	Number of times the local AS is to be prepended; an integer from 1-64.
<i>enforce-first-as</i>	Enforce the first AS for EBGP routes; one of <b>enable</b> , <b>disable</b> .
<i>fast-external-failover</i>	Immediately reset session if a link to a directly connected external peer goes down; one of <b>enable</b> , <b>disable</b> .
<i>log-neighbor-changes</i>	Log reasons for neighbor going up, down, and resetting; one of <b>enable</b> , <b>disable</b> .
<i>maxas-limit</i>	Allow the AS-PATH attribute from EBGP neighbor to impose a limit on the number of ASes; an integer from 0-2000.
<i>synchronization</i>	Perform IGP synchronization; one of <b>enable</b> , <b>disable</b> .

## Get BGP Best Path Configuration

Get the BGP best path configuration.

### Request

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

where:

Element	Description
<i>vrf_name</i>	VRF name; one of the VRF name, <b>default</b> , <b>all</b> . Default value: <b>default</b> .

### Response

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

where:

Element	Description
<i>vrf_name</i>	VRF name; one of the VRF name, <b>default</b> , <b>all</b> . Default value: <b>default</b> .
<i>always-compare-med</i>	Allow comparing MED from different neighbors; one of <b>enable</b> , <b>disable</b> .
<i>as-path-ignore</i>	Ignore as-path length in selecting a route; one of <b>enable</b> , <b>disable</b> .

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



## Get BGP Confederation Configuration

Get the BGP confederation configuration.

### Request

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

where:

Element	Description
<i>vrf_name</i>	VRF name; one of the VRF name, default, all. Default value: default.

### Response

Response Body (JSON)	<pre>[   {     "vrf_name": "&lt;vrf_name&gt;",     "identifier": "&lt;identifier&gt;",     "peers": "&lt;peers&gt;",   } ]</pre>
-------------------------	--

where:

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

## Get BGP Graceful-Restart Configuration

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

## Get BGP Route Reflector Information

Get BGP route reflector information.

### Request

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

where:

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

## Get BGP RIB Information

Get 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
<i>af_name</i>	(Optional) Address family name; one or both of ipv4, ipv6. Default value; both.
<i>vrf_name</i>	VRF name; one of the VRF name, default, all. Default value: default.

## Response

Response Body (JSON)	<pre>[   {     "status": "&lt;status&gt;",     "network": "&lt;network&gt;",     "nextHopGlobal": "&lt;nextHopGlobal&gt;",     "nextHopLocal": "&lt;nextHopLocal&gt;",     "Metric": "&lt;metric&gt;",     "local_pref": "&lt;local_pref&gt;",     "weight": "&lt;weight&gt;",     "pathInfo": "&lt;pathInfo&gt;",     "medvalue": "&lt;medvalue&gt;",     "med": "&lt;med&gt;",     "aspath": "&lt;as path string&gt;",     "aspath4B": "&lt;aspath4B&gt;",     "origin": "&lt;origin&gt;"   }, ]</pre>
-------------------------	--

where:

Element	Description
<i>status</i>	Router status code; one of: <ul style="list-style-type: none"> <li>● s - suppressed</li> <li>● d - damped</li> <li>● h - history</li> <li>● * - valid</li> <li>● &gt; - best</li> <li>● i - internal</li> </ul>
<i>network</i>	Route destination IP address; a valid IPv4 or IPv6 address.
<i>nextHopGlobal</i>	Route nexthop IPv6 address. Not used for IPv4.
<i>nextHopLocal</i>	Route nexthop; a valid IPv4 or IPv6 address.
<i>weight</i>	Route weight attribute; an integer from 0-65535.
<i>pathInfo</i>	Route path information; a valid AS path VTY string.
<i>medvalue</i>	Multi-exit discriminator value if the MED attribute is missing and missing-as-worst is set; an integer from 0-4294967294.
<i>med</i>	Multi-exit discriminator value; an integer from 0-4294967294.
<i>aspath</i>	Route AS path attribute; a valid AS path VTY string.
<i>aspath4B</i>	Route 4B AS path; a valid AS path VTY string.
<i>origin</i>	Route origin attribute; one of the following: <ul style="list-style-type: none"> <li>● i - IGP</li> <li>● e - EGP</li> <li>● ? - incomplete</li> </ul>

## Get BGP Detailed RIB Information

Get 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
<i>af_name</i>	(Optional) Address family name; one or both of <code>ipv4</code> , <code>ipv6</code> . Default value: <code>both</code> .
<i>route</i>	Route; a valid IPv4 or IPv6 address.
<i>network_mask</i>	Network mask: <ul style="list-style-type: none"><li>• IPv4: An integer from 0-32.</li><li>• IPv6: An integer from 0-128.</li></ul>
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .

## Response

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

where:

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



Element	Description
<i>stale</i>	Whether the state is stale; Yes. <b>Note:</b> This value only appears if the state is stale.
<i>Multipath-candidate</i>	Whether this is a multipath candidate; one of Yes, No.
<i>installed</i>	Whether installed; one of Yes, No.
<i>synchronized</i>	Whether synchronized; one of Yes, No.
<i>Atomic aggregate</i>	Whether this is an atomic aggregate; Yes. <b>Note:</b> This value only appears if it is Yes.
<i>best</i>	Whether this is the best path; one of Yes, No.
<i>community</i>	Community string.
<i>Extended community</i>	Extended community string.
<i>originator</i>	Originator ID.
<i>Cluster id</i>	Cluster ID.
<i>Reuse info</i>	Reuse information.
<i>Last update</i>	Last update time.
<i>best is no.</i>	Which path number is best; the maximum number of paths for this destination.
<i>advertised to any peer</i>	Whether this is advertised to any peers; one of Yes, No.
<i>advertised to EBGp peer</i>	Whether this is advertised to an EBGp peer; one of Yes, No.
<i>advertised outside local AS</i>	Whether this is advertised outside the local AS; one of Yes, No.
<i>advertisement s suppressed by an agregate</i>	Whether advertisements are suppressed by an aggregate; one of Yes, No.
<i>advertised to non peer-group peers</i>	IP address advertised to non peer-group peers.
<i>advertised to peer-groups</i>	IP address addvertised to peer groups.
<i>Not advertised</i>	Not advertised to any peer. <b>Note:</b> This value only appears if t rue.

## Get BGP Summary Information

Get 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
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, default, all. Default value: default.
<i>af_name</i>	(Optional) Address family name; one or both of ipv4, ipv6. Default value: ipv4.
<i>subaf_name</i>	Subsequent Address Family Identifier name; unicast. Default value: unicast.

### Response

Response Body (JSON)	<pre>[   {     "router id": "&lt;router id&gt;"     "peer": "&lt;peer&gt;"     "peer version": "&lt;peer version&gt;"     "peer AS": "&lt;peer AS&gt;"     "open in": "&lt;open in&gt;"     "update in": "&lt;update in&gt;"     "keepalive in": "&lt;keepalivein&gt;"     "refresh in": "&lt;refresh in&gt;"     "dynamic cap in": "&lt;dynamic cap in&gt;"     "open out": "&lt;open out&gt;"     "update out": "&lt;update out&gt;"     "keepalive out": "&lt;keepalive out&gt;"     "refresh out": "&lt;refresh out&gt;"     "dynamic cap out": "&lt;dynamic cap out&gt;"   }, ]</pre>
-------------------------	--

where:

Element	Description
<i>router id</i>	Router ID; a valid IPv4 or IPv6 address.
<i>peer</i>	Peer address; a valid IPv4 or IPv6 address.

<b>Element</b>	<b>Description</b>
<i>peer version</i>	Peer version.
<i>peer AS</i>	Peer AS.
<i>open in</i>	Number of received open messages.
<i>update in</i>	Number of received updates.
<i>keepalive in</i>	Number of received keepalives.
<i>refresh in</i>	Number of received route refresh.
<i>dynamic cap in</i>	Dynamic capabilities input count.
<i>open out</i>	Number of sent open messages.
<i>update out</i>	Number of sent updates.
<i>keepalive out</i>	Number of sent keepalive messages.
<i>refresh out</i>	Number of sent route refresh messages.
<i>dynamic cap out</i>	Dynamic capabilities output count.

## Get BGP Neighbor Details

Get 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
<i>nbr-ip</i>	(Optional) Neighbor IP address; a valid IPv4 or IPv6 address.
<i>vrf_name</i>	(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": "&lt;uptime&gt;" "peer-group name": "&lt;peer-group name&gt;" "holdtime": "&lt;holdtime&gt;" "keepalive": "&lt;keepalive&gt;" "conf holdtime": "&lt;conf holdtime&gt;" "conf keepalive": "&lt;conf keepalive&gt;" "recvMsg": "&lt;recvMsg&gt;" "recvNotf": "&lt;recvNotf&gt;" "recvQueue": "&lt;recvQueue&gt;" "sentMsg": "&lt;sentMsg&gt;" "sentNotf": "&lt;sentNotf&gt;" "sentQueue": "&lt;sentQueue&gt;" "refresh_in": "&lt;refresh_in&gt;" "refresh_out": "&lt;refresh_out&gt;" "routeadv": "&lt;routeadv&gt;" "update_if": "&lt;update_if&gt;" "update_source": "&lt;update_source&gt;" "established": "&lt;established&gt;" "dropped": "&lt;dropped&gt;" "prefix overflow": "&lt;prefix overflow&gt;" "ttl": "&lt;ttl&gt;" "local address": "&lt;local address&gt;" "local port": "&lt;local port&gt;" "remote address": "&lt;remote address&gt;" "remote port": "&lt;remote port&gt;" "nextHopAddress": "&lt;nextHopAddress&gt;" "nextHopLocalV6": "&lt;nextHopLocalV6&gt;" "nextHopGlobalV6": "&lt;nextHopGlobalV6&gt;" "shared_network": "&lt;shared_network&gt;" "next conn retry": "&lt;next conn retry&gt;" "err notif": "&lt;err notif&gt;" "last_reset_time": "&lt;last_reset_time&gt;" "error code": "&lt;error code&gt;" "error subcode": "&lt;error subcode&gt;" "rmap_map": "&lt;rmap_map&gt;"     }, ] </pre>
-----------------------------------	--

where:

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

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

<b>Element</b>	<b>Description</b>
<i>recvNotf</i>	Number of received notifications.
<i>recvQueue</i>	Received messages queue count.
<i>sentMsg</i>	Number of sent messages.
<i>sentNotf</i>	Number of sent notifications.
<i>sentQueue</i>	Sent messages queue count.
<i>refresh_in</i>	Number of route refresh messages received.
<i>refresh_out</i>	Number of route refresh messages sent.
<i>routeadv</i>	Number of router advertisements.
<i>update_if</i>	Update interface.
<i>update_source</i>	Update source address.
<i>established</i>	Established count.
<i>dropped</i>	Dropped count.
<i>prefix overflow</i>	Whether there is a prefix overflow; one of Yes, No.
<i>ttl</i>	Time to live value.
<i>local address</i>	Local neighbor IP address.
<i>local port</i>	Local port number.
<i>remote address</i>	Remote peer IP address.
<i>remote port</i>	Remote port number.
<i>nextHop Address</i>	Next-hop address.
<i>nextHopLocal V6</i>	Next-hop address (link local).
<i>nextHop GlobalV6</i>	Next-hop address (global).
<i>shared network</i>	Shared network.
<i>next conn retry</i>	Number of retries.
<i>err notif</i>	Whether there was an error notification; one of sent, received.
<i>last_reset_ time</i>	Last reset time.
<i>err code</i>	Code string.

<b>Element</b>	<b>Description</b>
<i>err_subcode</i>	Subcode string.
<i>rmap_name</i>	Default originating route map.



## Get BGP Neighbor Statistics

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

### Response

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

where:

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

## Get BGP Distance Configuration

Get 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
<i>af_name</i>	Address family name; one or both of <code>ipv4</code> , <code>ipv6</code> . Default value: <code>both</code> .
<i>saf_name</i>	Subsequent Address Family Identifier name; <code>unicast</code> . Default value: <code>unicast</code> .
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .

### Response

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

where:

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

## Get BGP Address Family Global Configuration

Get 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
<i>af_name</i>	Address family name; one or both of ipv4, ipv6. Default value; both.
<i>saf_name</i>	Subsequent Address Family Identifier name; unicast. Default value: unicast.
<i>vrf_name</i>	(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
<i>vrf_name</i>	VRF name; one of the VRF name, default, all. Default value: default.
<i>cc_reflection</i>	Client-to-client reflect; one of enable, disable.
<i>synchronization</i>	Perform IGP synchronization; one of enable, disable.
<i>network_synchronization</i>	Perform IGP synchronization on network routes; one of enable, disable.

## Get BGP Multipath ECMP Numbers Configuration

Get 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
<i>af_name</i>	Address family name; one or both of <code>ipv4</code> , <code>ipv6</code> . Default value: <code>both</code> .
<i>saf_name</i>	Subsequent Address Family Identifier name; <code>unicast</code> . Default value: <code>unicast</code> .
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .

### Response

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

where:

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

## Get BGP Nexthop Trigger-Delay Configuration

Get 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
<i>af_name</i>	Address family name; one or both of ipv4, ipv6. Default value; both.
<i>saf_name</i>	Subsequent Address Family Identifier name; unicast. Default value: unicast.

### Response

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

where:

Element	Description
<i>critical</i>	Nexthop changes affecting reachability; an integer from 1-4294967295.
<i>non-critical</i>	Nexthop changes affecting metric; an integer from 1-4294967295.

## Get BGP Aggregate Configuration

Get 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
<i>af_name</i>	Address family name; one or both of <code>ipv4</code> , <code>ipv6</code> . Default value: <code>both</code> .
<i>saf_name</i>	Subsequent Address Family Identifier name; <code>unicast</code> . Default value: <code>unicast</code> .
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .

### Response

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

where:

Element	Description
<i>vrf_name</i>	VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .
<i>prefix</i>	Aggregate prefix; an IP address in one of the following forms: <ul style="list-style-type: none"><li>• A.B.C.D/M</li><li>• X:X::X:X/M.</li></ul>
<i>type</i>	Aggregate type; one of the following: <ul style="list-style-type: none"><li>• <code>as_set</code> - Generate AS set path information.</li><li>• <code>summary_only</code> - Filter more specific routes from updates.</li><li>• <code>as_set_summary_only</code> - Both <code>as-set</code> and <code>summary-only</code>.</li></ul>

## Get BGP Dampening Parameters Configuration

Get 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
<i>af_name</i>	Address family name; one or both of ipv4, ipv6. Default value; both.
<i>saf_name</i>	Subsequent Address Family Identifier name; unicast. Default value: unicast.
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, default, all. Default value: default.

### Response

Response Body (JSON)	<pre>[   {     "vrf_name ": "&lt;vrf_name&gt;",     "half_life ": "&lt;half_life&gt;",     "reuse_penalty ": "&lt;reuse_penalty&gt;",     "suppress_penalty ": "&lt;suppress_penalty&gt;",     "max_suppress ": "&lt;max_suppress&gt;",     "unreach_half_life ": "&lt;unreach_half_life&gt;",     "rmap_name ": "&lt;rmap_name&gt;"   } ]</pre>
-------------------------	--

where:

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

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



## Get BGP Dampened Path Configuration

Get 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
<i>af_name</i>	Address family name; one or both of ipv4, ipv6. Default value; both.
<i>saf_name</i>	Subsequent Address Family Identifier name; unicast. Default value: unicast.
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, default, all. Default value: default.

### Response

Response Body (JSON)	<pre>[   {     "statusCode": "&lt;statusCodeVal&gt;",     "network": "&lt;networkVal&gt;",     "nextHopGlobal": "&lt;nextHopGlobalVal&gt;",     "nextHopLocal": "&lt;nextHopLocalVal&gt;",     "metric": "&lt;metricVal&gt;",     "pathInfo": "&lt;pathInfoVal&gt;",     "reuseTime": "&lt;reuseTimeVal&gt;",     "asPathStr": "&lt;asPathStrVal&gt;",     "asPath4BStr": "&lt;asPath4BStrVal&gt;",     "routeOriginType": "&lt;routeOriginTypeVal&gt;"   } ]</pre>
-------------------------	---

where:

<b>Element</b>	<b>Description</b>
<i>statusCode</i>	Router status code; one of: <ul style="list-style-type: none"><li>● s - suppressed</li><li>● d - damped</li><li>● h - history</li><li>● * - valid</li><li>● &gt; - best</li><li>● i - internal</li></ul>
<i>network</i>	Route destination IP address; a valid IPv4 or IPv6 address.
<i>nextHopGlobal</i>	Route nexthop IPv6 address. Not used for IPv4.
<i>nextHopLocal</i>	Route nexthop; a valid IPv4 or IPv6 address.
<i>metric</i>	Route metric; an integer.
<i>pathInfo</i>	Route path information; a valid AS path VTY string.
<i>reuseTime</i>	Route reuse time; a string.
<i>asPathStr</i>	Route AS path attribute; a valid AS path VTY string.
<i>asPath4BStr</i>	Route 4B AS path; a valid AS path VTY string.
<i>routeOrigin Type</i>	Route origin type; one of the following: <ul style="list-style-type: none"><li>● i - IGP</li><li>● e - EGP</li><li>● ? - incomplete</li></ul>

## Get BGP Network Configuration

Get 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
<i>af_name</i>	Address family name; one or both of <code>ipv4</code> , <code>ipv6</code> . Default value; both.
<i>saf_name</i>	Subsequent Address Family Identifier name; <code>unicast</code> . Default value: <code>unicast</code> .
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .

### Response

Response Body (JSON)	<pre>[   {     "vrf_name ": "&lt;vrf_name&gt;",     "prefix ": "&lt;prefix&gt;",     "backdoor": "&lt;backdoor&gt;",     "rmap_name": "&lt;rmap_name&gt;"   } ]</pre>
-------------------------	---

where:

Element	Description
<i>vrf_name</i>	VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .
<i>prefix</i>	Network prefix; an IP address in one of the following forms: <ul style="list-style-type: none"><li>● A.B.C.D/M</li><li>● X:X::X:X/M.</li></ul>
<i>backdoor</i>	Whether a BGP backdoor route is specified; one of <code>enable</code> , <code>disable</code> .
<i>rmap_name</i>	Route map name; a string up to 63 characters long.

## Get BGP Redistribute Configuration

Get 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
<i>af_name</i>	Address family name; one or both of <code>ipv4</code> , <code>ipv6</code> . Default value: <code>both</code> .
<i>saf_name</i>	Subsequent Address Family Identifier name; <code>unicast</code> . Default value: <code>unicast</code> .
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .

### Response

Response Body (JSON)	<pre>[   {     "vrf_name ": "&lt;vrf_name&gt;",     "redist_direct": "&lt;redist_direct&gt;",     "direct_rmap_name ": "&lt;direct_rmap_name&gt;",     "redist_ospf": "&lt;redist_ospf&gt;",     "ospf_rmap_name ": "&lt;ospf_rmap_name&gt;",     "redist_static ": "&lt;redist_static&gt;",     "static_rmap_name ": "&lt;static_rmap_name&gt;"   } ]</pre>
-------------------------	--

where:

Element	Description
<i>vrf_name</i>	VRF name; one of the VRF name, <code>default</code> , <code>all</code> . Default value: <code>default</code> .
<i>redist_direct</i>	Whether redistribute direct is enabled; one of <code>enable</code> , <code>disable</code> .
<i>direct_rmap_name</i>	Route map name for redistribute direct; a string up to 63 characters long.
<i>redist_ospf</i>	Whether redistribute OSPF is enabled; one of <code>enable</code> , <code>disable</code> .

<b>Element</b>	<b>Description</b>
<i>ospf_rmap_name</i>	Route map name for redistribute OSPF; a string up to 63 characters long.
<i>redist_static</i>	Whether redistribute static is enabled; one of <b>enable</b> , <b>disable</b> .
<i>static_rmap_name</i>	Route map name for redistribute static; a string up to 63 characters long.

## Get BGP Neighbor Configuration

Get 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
<i>neighbor_ip</i>	(Optional) The IP address of the neighbor; a valid IPv4 or IPv6 address. No value will display all neighbors.
<i>vrf_name</i>	(Optional) VRF name; one of the VRF name, default, all. Default value: default.

## Response

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

where:

Element	Description
<i>neighbor</i>	Neighbor IP address; a valid IPv4 or IPv6 address.
<i>vrf_name</i>	VRF name; one of the VRF name, default, all. Default value: default.

<b>Element</b>	<b>Description</b>
<i>remote as</i>	Current neighbor AS; an AS number.
<i>local as</i>	Switch AS; an AS number.
<i>address family</i>	Neighbor address family; an address family.
<i>advertisement interval</i>	Minimum interval between BGP updates, in seconds; an integer.
<i>bfd</i>	BFD state; one of: <ul style="list-style-type: none"> <li>● enabled</li> <li>● disabled</li> <li>● multihop enabled</li> </ul>
<i>connection retry time</i>	BGP connect timer, in seconds; an integer.
<i>description</i>	Neighbor description; a string.
<i>disallow infinite holdtime</i>	Neighbor disallow infinite hold time; one of Yes, No.
<i>do not capability negotiate</i>	Whether to perform capability negotiations; one of Yes, No.
<i>advertise dynamic capability</i>	Advertise dynamic capability to this neighbor; one of Yes, No.
<i>EBGP multihop</i>	Number of multihops; an integer from 1 - 255.
<i>remote private as</i>	Whether to remove private AS number from outbound packets; one of Yes, No.
<i>maximum peers</i>	Maximum number of peers for this prefix; an integer from 1-96.
<i>password</i>	Neighbor password; an encrypted password.
<i>shutdown</i>	Neighbor state; one of Yes, No.
<i>peer holdtime</i>	Holdtime value, in seconds; an integer from 0-3600.
<i>peer keepalive</i>	Keepalive value, in seconds; an integer from 0-3600.
<i>connection-mode passive</i>	Whether to allow a passive connection; one of Yes, No.
<i>tll security hops</i>	Number of hops; an integer from 1-254.



Element	Description
<i>update-source</i>	Source of routing updates; one of: <ul style="list-style-type: none"> <li>● ethernet</li> <li>● vlan</li> <li>● loopback interfaces</li> </ul>
<i>weight</i>	The default weight for routes from this neighbor; an integer from 0–65535.
<i>allow as in</i>	Accept AS path with own AS in it; an integer from 1-10.
<i>default originate</i>	Whether to originate default route to this neighbor; one of Yes, No.
<i>default originate rmap</i>	Route map that specifies criteria; an RMAP name.
<i>prefix-list in</i>	Filter updates from this neighbor; a filter name.
<i>prefix-list out</i>	Filter updates to this neighbor; a filter name.
<i>maximum-prefix</i>	Maximum number of prefix accept from this peer; an integer from -15782.
<i>maximum-prefix warning</i>	Whether to only give a warning message when limit is exceeded; one of Yes, No.
<i>maximum-prefix threshold percent</i>	Threshold value; an integer from 1-100.
<i>next-hop-self</i>	Whether to disable the next hop calculation for this neighbor; one of Yes, No.
<i>filter-list in</i>	Establish filter for incoming routes; a filter name.
<i>filter-list out</i>	Establish filter for outgoing routes; a filter name.
<i>route-map in</i>	Apply routemap for incoming routes; an RMAP name.
<i>route-map out</i>	Apply routemap for outgoing routes; an RMAP name.
<i>route reflector client</i>	Whether to set neighbor as route reflector client; one of Yes, No.
<i>send community</i>	Whether to send community attribute to this neighbor; one of Yes, No.
<i>send community extended</i>	Whether to send extended community attribute to this neighbor; one of Yes, No.

<b>Element</b>	<b>Description</b>
<i>soft reconfiguration inbound</i>	Whether to allow inbound soft reconfiguration for this neighbor; one of Yes, No.
<i>unsuppress-map</i>	Route map to selectively unsuppress suppressed routes; an RMAP name.

---

## 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

The following STP interface property commands are available:

- [Get STP Properties for All Interfaces](#)
- [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 the Global DHCP Client Feature Property

Get 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
<i>ena_dhcp_feature</i>	Whether the DHCP client feature is enabled globally; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .  <b>Note:</b> 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

Set 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
<i>ena_dhcp_feature</i>	Enable or disable the DHCP client feature globally; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .  <b>Note:</b> 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
<i>ena_dhcp_feature</i>	Whether the DHCP client feature is enabled globally; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .  <b>Note:</b> 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

Get 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)	<pre>{   "if_name": "&lt;if_name&gt;",   "ena_v4_client": "&lt;ena_v4_client&gt;",   "ena_v6_client": "&lt;ena_v6_client&gt;",   "req_hostname": "&lt;req_hostname&gt;",   "req_ntp_server": "&lt;req_ntp_server&gt;",   "req_log_server": "&lt;req_log_server&gt;",   "class_id": "&lt;class_id&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>ena_v4_client</i>	Whether the DHCPv4 client is enabled on the interface; one of Yes, No. Default value: Yes.
<i>ena_v6_client</i>	Whether the DHCPv6 client is enabled on the interface; one of Yes, No. Default value: Yes.
<i>req_hostname</i>	Whether a request has been issued for the host name option on an interface; one of Yes, No. Default value: Yes.
<i>req_ntp_server</i>	Whether a request has been issued for the NTP-server option on an interface; one of Yes, No. Default value: Yes.
<i>req_log_server</i>	Whether a request has been issued for the Log server option on an interface; one of Yes, No. Default value: Yes.
<i>class_id</i>	The name of Vendor class identifier. <b>Note:</b> The Vendor class identifier name must exist.

## Update DHCP Client Interface Properties

Get DHCP client properties of a specific interface.

### Request

Method Type	PUT
Request URI	/nos/api/cfg/dhcp_client/interface/<if_name>
Request Body (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "ena_v4_client": "&lt;ena_v4_client&gt;",   "ena_v6_client": "&lt;ena_v6_client&gt;",   "req_hostname": "&lt;req_hostname&gt;",   "req_ntp_server": "&lt;req_ntp_server&gt;",   "req_log_server": "&lt;req_log_server&gt;",   "class_id": "&lt;class_id&gt;" }</pre>

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>ena_v4_client</i>	Whether the DHCPv4 client is enabled on the interface; one of Yes, No. Default value: Yes.
<i>ena_v6_client</i>	Whether the DHCPv6 client is enabled on the interface; one of Yes, No. Default value: Yes.
<i>req_hostname</i>	Whether a request has been issued for the host name option on an interface; one of Yes, No. Default value: Yes.
<i>req_ntp_server</i>	Whether a request has been issued for the NTP-server option on an interface; one of Yes, No. Default value: Yes.
<i>req_log_server</i>	Whether a request has been issued for the Log server option on an interface; one of Yes, No. Default value: Yes.
<i>class_id</i>	The name of Vendor class identifier. <b>Note:</b> The Vendor class identifier name must exist.

## Response

Response Body (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "ena_v4_client": "&lt;ena_v4_client&gt;",   "ena_v6_client": "&lt;ena_v6_client&gt;",   "req_hostname": "&lt;req_hostname&gt;",   "req_ntp_server": "&lt;req_ntp_server&gt;",   "req_log_server": "&lt;req_log_server&gt;",   "class_id": "&lt;class_id&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>ena_v4_client</i>	Whether the DHCPv4 client is enabled on the interface; one of Yes, No. Default value: Yes.
<i>ena_v6_client</i>	Whether the DHCPv6 client is enabled on the interface; one of Yes, No. Default value: Yes.
<i>req_hostname</i>	Whether a request has been issued for the host name option on an interface; one of Yes, No. Default value: Yes.
<i>req_ntp_server</i>	Whether a request has been issued for the NTP-server option on an interface; one of Yes, No. Default value: Yes.
<i>req_log_server</i>	Whether a request has been issued for the Log server option on an interface; one of Yes, No. Default value: Yes.
<i>class_id</i>	The name of Vendor class identifier. <b>Note:</b> The Vendor class identifier name must exist.



## Delete the Vendor Class Identifier of an Interface

Delete 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
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.

### *Response*

Response Body (JSON)	
-------------------------	--

## Get the Global DHCP Relay Service Property

Get 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
<i>ena_v4_relay</i>	Whether DHCPv4 relay is enabled on the interface; one of Yes, No. Default value: Yes. <b>Note:</b> 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.
<i>ena_v6_relay</i>	Whether DHCPv6 relay is enabled on the interface; one of Yes, No. Default value: Yes. <b>Note:</b> 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

Update 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
<i>ena_v4_relay</i>	Whether DHCPv4 relay is enabled on the interface; one of Yes, No. Default value: Yes. <b>Note:</b> 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.
<i>ena_v6_relay</i>	Whether DHCPv6 relay is enabled on the interface; one of Yes, No. Default value: Yes. <b>Note:</b> 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

Get 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)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "dhcpv4_relay":     [       {         "v4_relay_addr": "&lt;v4_relay_addr&gt;",       }     ],     "dhcpv6_relay":     [       {         "v6_relay_addr": "&lt;v6_relay_addr&gt;",         "v6_relay_out_if": "&lt;v6_relay_out_if&gt;"       }     ]   } ]</pre>
-------------------------	--

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>v4_relay_addr</i>	IPv4 address of the relay server; a valid IPv4 address.
<i>v6_relay_addr</i>	IPv6 address of the relay server; a valid IPv6 address.
<i>v6_relay_out_if</i>	Outgoing interface of the relay service. <b>Note:</b> The interface must exist.

## Get DHCP Relay Interface Properties

Get 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
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.

### Response

Response Body (JSON)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "dhcpv4_relay":       [         {           "v4_relay_addr": "&lt;v4_relay_addr&gt;",         }       ],     "dhcpv6_relay":       [         {           "v6_relay_addr": "&lt;v6_relay_addr&gt;",           "v6_relay_out_if": "&lt;v6_relay_out_if&gt;"         }       ]   } ]</pre>
-------------------------	--

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>v4_relay_addr</i>	IPv4 address of the relay server; a valid IPv4 address.
<i>v6_relay_addr</i>	IPv6 address of the relay server; a valid IPv6 address.
<i>v6_relay_out_if</i>	Outgoing interface of the relay service. <b>Note:</b> 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

Update 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)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "dhcpv4_relay":       [         {           "v4_relay_addr": "&lt;v4_relay_addr&gt;",         }       ],     "dhcpv6_relay":       [         {           "v6_relay_addr": "&lt;v6_relay_addr&gt;",           "v6_relay_out_if": "&lt;v6_relay_out_if&gt;"         }       ]   } ]</pre>

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>v4_relay_addr</i>	IPv4 address of the relay server; a valid IPv4 address.
<i>v6_relay_addr</i>	IPv6 address of the relay server; a valid IPv6 address.
<i>v6_relay_out_if</i>	Outgoing interface of the relay service. <b>Note:</b> The interface must exist.

## Response

Response Body (JSON)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "dhcpv4_relay":     [       {         "v4_relay_addr": "&lt;v4_relay_addr&gt;",       }     ],     "dhcpv6_relay":     [       {         "v6_relay_addr": "&lt;v6_relay_addr&gt;",         "v6_relay_out_if": "&lt;v6_relay_out_if&gt;"       }     ]   } ]</pre>
-------------------------	--

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>v4_relay_addr</i>	IPv4 address of the relay server; a valid IPv4 address.
<i>v6_relay_addr</i>	IPv6 address of the relay server; a valid IPv6 address.
<i>v6_relay_out_if</i>	Outgoing interface of the relay service. <b>Note:</b> The interface must exist.



## Delete DHCP Relay Interface Properties

Delete 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)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "dhcpv4_relay":       [         {           "v4_relay_addr": "&lt;v4_relay_addr&gt;",         }       ],     "dhcpv6_relay":       [         {           "v6_relay_addr": "&lt;v6_relay_addr&gt;",           "v6_relay_out_if": "&lt;v6_relay_out_if&gt;"         }       ]   } ]</pre>

where:

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

## Response

Response Body (JSON)	<pre>[   {     "if_name": "&lt;if_name&gt;",     "dhcpv4_relay":     [       {         "v4_relay_addr": "&lt;v4_relay_addr&gt;",       }     ],     "dhcpv6_relay":     [       {         "v6_relay_addr": "&lt;v6_relay_addr&gt;",         "v6_relay_out_if": "&lt;v6_relay_out_if&gt;"       }     ]   } ]</pre>
-------------------------	--

where:

Element	Description
<i>if_name</i>	Interface name. <b>Note:</b> The interface must exist.
<i>v4_relay_addr</i>	IPv4 address of the relay server; a valid IPv4 address.
<i>v6_relay_addr</i>	IPv6 address of the relay server; a valid IPv6 address.
<i>v6_relay_out_if</i>	Outgoing interface of the relay service. <b>Note:</b> The interface must exist.

---

## STP

The following Spanning Tree Protocol (STP) URIs are available:

- /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 commands are available:

- [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 Per VLAN

Get 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
<i>vlan_ID</i>	VLAN ID; an integer from 2-3999.

### Response

Response Body (JSON)	<pre>{   "forward-time": "&lt;forward-time&gt;",   "hello-time": "&lt;hello-time&gt;",   "max-age" : "&lt;max-age&gt;",   "priority" : "&lt;priority&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>forward-time</i>	The forward delay for the spanning tree; an integer from 4-30.
<i>hello-time</i>	The hello interval for the spanning tree; an integer from 1-10.
<i>max-age</i>	The maximum age interval for the spanning tree; an integer from 6-40.
<i>priority</i>	The bridge priority for the spanning tree; an integer from 0-61440.

## Set STP Properties Per VLAN

Set 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
<i>vlan_ID</i>	VLAN number.; an integer from 2-3999.
<i>forward-time</i>	The forward delay for the spanning tree; an integer from 4-30.
<i>hello-time</i>	The hello interval for the spanning tree; an integer from 1-10.
<i>max-age</i>	The maximum age interval for the spanning tree; an integer from 6-40.
<i>priority</i>	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
<i>forward-time</i>	(Optional) The forward delay for the spanning tree; an integer from 4-30.
<i>hello-time</i>	(Optional) The hello interval for the spanning tree; an integer from 1-10.
<i>max-age</i>	(Optional) The maximum age interval for the spanning tree; an integer from 6-40.
<i>priority</i>	(Optional) The bridge priority for the spanning tree; an integer from 0-61440.

## Get STP Interface Properties

Get 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
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.

### Response

Response Body (JSON)	<pre>{   "if_name": "&lt;if_name&gt;",   "edge_port": "&lt;edge_port&gt;",   "bpdu_guard": "&lt;bpdu_guard&gt;",   "loop_guard": "&lt;loop_guard&gt;",   "root_guard": "&lt;root_guard&gt;" }</pre>
-------------------------	---

where:

Element	Description
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>edge_port</i>	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .
<i>bpdu_guard</i>	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>loop_guard</i>	(Optional) Whether loop guard is enabled on a port for additional checks for preventing STP looping; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>root_guard</i>	(Optional) Whether guard mode is set to root guard on interface.

## Update STP Interface Properties

Update 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)	<pre>{   "if_name": "&lt;if_name&gt;",   "edge_port": "&lt;edge_port&gt;",   "bpdu_guard": "&lt;bpdu_guard&gt;",   "loop_guard": "&lt;loop_guard&gt;",   "root_guard": "&lt;root_guard&gt;" }</pre>

where:

Element	Description
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>edge_port</i>	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of <b>yes</b> , <b>no</b> . Default value: <b>yes</b> .
<i>bpdu_guard</i>	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>loop_guard</i>	(Optional) Whether look guard is enabled on a port for additional checks for preventing STP looping; one of <b>enable</b> , <b>disable</b> . Default value: <b>disable</b> .
<i>root_guard</i>	(Optional) Whether guard mode is set to root guard on interface.

## Get STP Interface VLAN Properties

Get 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
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>vlan_ID</i>	VLAN ID; an integer from 2-3999.

### Response

Response Body (JSON)	{ "cost": "<cost>", "priority": "<priority>", }
-------------------------	--

where:

Element	Description
<i>cost</i>	The interface's spanning-tree port path cost; one of auto (based on port speed), an integer from 1-200000000.
<i>priority</i>	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

Update 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
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>vlan_ID</i>	VLAN ID; an integer from 2-3999.
<i>cost</i>	The interface's spanning-tree port path cost; one of <code>auto</code> (based on port speed), an integer from 1-200000000.
<i>priority</i>	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
<i>if_name</i>	The IP interface name; a string. <b>Note:</b> The interface must exist.
<i>edge_port</i>	Whether the interface is configured as an edge port, which allows the port to automatically transition to the STP forwarding state; one of <code>yes</code> , <code>no</code> . Default value: <code>yes</code> .
<i>bpdu_guard</i>	(Optional) Whether BPDU guard is enabled on a port, which automatically shuts down the interface upon receipt of a BPDU; one of <code>enable</code> , <code>disable</code> . Default value: <code>disable</code> .



---

## 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 [IBM ServerProven website](#) to make sure that the hardware and software is supported by your product.
- Go to the [IBM 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	<ul style="list-style-type: none"> <li>The room air must be continuously filtered with 40% atmospheric dust spot efficiency (MERV 9) according to ASHRAE Standard 52.2<sup>1</sup>.</li> <li>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.</li> <li>The deliquescent relative humidity of the particulate contamination must be more than 60%<sup>2</sup>.</li> <li>The room must be free of conductive contamination such as zinc whiskers.</li> </ul>
Gaseous	<ul style="list-style-type: none"> <li>Copper: Class G1 as per ANSI/ISA 71.04-1985<sup>3</sup></li> <li>Silver: Corrosion rate of less than 300 Å in 30 days</li> </ul>

<sup>1</sup> 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.

<sup>2</sup> 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.

<sup>3</sup> 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 EMC Directive Conformance Statement

This product is in conformity with the protection requirements of EU Council Directive 2004/108/EC 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 Information Technology Equipment according to European Standard EN 55022. The limits for Class A equipment were derived for commercial and industrial environments to provide reasonable protection against interference with licensed communication equipment.

Lenovo, Einsteinova 21, 851 01 Bratislava, Slovakia

## Germany Class A Statement

**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 EG Richtlinie 2004/108/EC (früher 89/336/EWG), 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 Paragraph 5 des EMVG ist die Lenovo (Deutschland) GmbH, Gropiusplatz 10, D-70563 Stuttgart.

Informationen in Hinsicht EMVG Paragraph 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: Dieses Produkt entspricht den Schutzanforderungen der EU-Richtlinie 2004/108/EG (früher 89/336/EWG) zur Angleichung der Rechtsvorschriften über die elektromagnetische Verträglichkeit in den EU-Mitgliedsstaaten und hält die Grenzwerte der EN 55022 Klasse A ein.

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 Betriebsmitteln**

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 2004/108/EG (früher 89/336/EWG) 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 EG Richtlinie 2004/108/EC (früher 89/336/EWG), 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 Paragraph 5 des EMVG ist die Lenovo (Deutschland) GmbH, Gropiusplatz 10, D-70563 Stuttgart.

Informationen in Hinsicht EMVG Paragraph 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 Genehmigungsverfahren ist 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。