System and IP Configuration APIs

System Configuration APIs

Table 1: System Configuration APIs

Action	Method	Payload Required	API
To retrieve complete information on system configuration	GET	No	/api/operational/system/settings-native /api/config/system/settings
To configure the system by setting the default gateway, management IP address and/or WAN IP address	PUT	Yes	/api/config/system/settings

Example for System Configuration Payload

```
<system>
 <settings>
  <hostname>MyNFVIS123</hostname>
  <mgmt>
   <ip>
    <address>192.168.1.2</address>
    <netmask>255.255.255.0</netmask>
   </ip></mgmt>
  <wan>
   <dhcp/>
  </wan>
  </settings>
</system>
```

Note In the example, the management interface is configured with a static IP address and the WAN interface is set to DHCP. You can configure both the management and the WAN interface with static IP addresses; however, you can configure DHCP on only one of the interfaces.

Table 2: Description for System Details Payload

Property	Type	Description	Mandatory/Default Value
hostname	String	Hostname of the system. The hostname now follows RFC952 rules, allowing only alphabets, numbers and hyphen. The hostname can begin and end with either an alphabet or a digit. Host software must handle host names of up to 255 characters.	Yes
default-gw	String	IP address of the default gateway. **Note** The default gateway assigned through the DHCP configuration will take precedence over the default gateway for static configuration. Hence, to use the default gateway for static configuration, disable DHCP configuration for the WAN interface. When using default gateway, DHCP configuration is not allowed on any interface, include WAN and MGMT interfaces.	Yes

mgmt ip address	String	Management IP address **Note** When an interface is configured with a static IP address, DHCP is automatically disabled on that interface.	Yes
mgmt ip netmask	String	Netmask for the IP address.	Yes
wan dhcp	String	Set dhcp on the WAN interface. **Note** You can configure DHCP either on the WAN interface or the management interface; you cannot configure DHCP on both the interfaces simultaneously.	No

Example: PUT System Configuration API

```
curl -v -u admin:admin -H "Accept:application/vnd.yang.data+xml" -H
"Content-Type:application/vnd.yang.data+xml" -k -X PUT
https://209.165.201.1/api/config/system -d
"<system>
 <settings>
  <hostname>Do3rdENCS75SettingsNoGW</hostname>
  <default-gw>172.19.183.1</default-gw>
  <mgmt>
   <ip>
    <address>172.19.183.75</address>
    <netmask>255.255.255.0</netmask>
   </ip>
  </mgmt>
  <wan>
   <ip>
    <address>4.3.2.5</address>
    <netmask>255.255.0.0</netmask>
   </ip><
  /wan>
 </settings>
</system>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (172.19.183.75) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
```

```
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Sep  2 17:03:09 2016 GMT
*  expire date: Aug 31 17:03:09 2026 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> PUT /api/config/system HTTP/1.1
> Host: 172.19.183.75
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.50.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 281
>
* upload completely sent off: 281 out of 281 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.6.3
< Date: Wed, 07 Sep 2016 02:43:26 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Wed, 07 Sep 2016 02:43:25 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1473-216205-877863
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
sj22lab-as1:149>
```

Example: GET System Details API

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X GET
https://209.165.201.1/api/operational/system/settings-native
Note: Unnecessary use of -X or --request, GET is already inferred.
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Sep  2 17:03:09 2016 GMT
*  expire date: Aug 31 17:03:09 2026 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/operational/system/settings-native HTTP/1.1
> Host: 172.19.183.75
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.50.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
>
< HTTP/1.1 200 OK
< Server: nginx/1.6.3
< Date: Tue, 06 Sep 2016 20:35:13 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
<settings-native xmlns="http://www.cisco.com/nfv" xmlns:y="http://tail-f.com/ns/rest"
xmlns:system="http://www.cisco.com/nfv">
  <mgmt>
   <ip-info>
      <interface>MGMT</interface>
      <ipv4_address>192.168.1.2</ipv4_address>
      <netmask>255.255.255.0</netmask>
      <ipv6_address>fe80::2f2:8bff:fec3:4a54</ipv6_address>
      <prefixlen>64</prefixlen>
      <mac_address>00:f2:8b:c3:4a:54</mac_address>
      <mtu>1500</mtu>
      <txqueuelen>1000</txqueuelen>
    </ip-info>
    <stats>
      <rx_packets>12481280</rx_packets>
      <rx_bytes>14392431432</rx_bytes>
      <rx_errors>0</rx_errors>
      <rx_dropped>210</rx_dropped>
      <rx_overruns>0</rx_overruns>
      <rx_frame>0</rx_frame>
      <tx_packets>3080505</tx_packets>
      <tx_bytes>238975886</tx_bytes>
      <tx_errors>0</tx_errors>
      <tx_dropped>0</tx_dropped>
      <tx_overruns>0</tx_overruns>
      <tx_carrier>0</tx_carrier>
      <tx_collisions>0</tx_collisions>
    </stats>
    <dhcp>
      <enabled>false</enabled>
      <offer>false</offer>
      <interface>NA</interface>
      <fixed_address>0.0.0.0</fixed_address>
      <subnet_mask>0.0.0.0</subnet_mask>
      <gateway>0.0.0.0</gateway>
      <lease_time>0</lease_time>
      <message_type>0</message_type>
      <name_servers>NA</name_servers>
      <server_identifier>0.0.0.0</server_identifier>
      <renewal_time>0</renewal_time>
      <rebinding_time>0</rebinding_time>
      <vendor_encapsulated_options>NA</vendor_encapsulated_options>
      <domain_name>NA</domain_name>
      <renew>0001-01-01T00:00:00-00:00</renew>
      <rebind>0001-01-01T00:00:00-00:00</rebind>
      <expire>0001-01-01T00:00:00-00:00</expire>
    </dhcp>
  </mgmt>
  <wan>
    <ip-info>
      <interface>wan-br</interface>
      <ipv4_address>209.165.201.22</ipv4_address>
      <netmask>255.255.255.0</netmask>
      <ipv6_address>fe80::2f2:8bff:fec3:49e0</ipv6_address>
      <prefixlen>64</prefixlen>
      <mac_address>00:f2:8b:c3:49:e0</mac_address>
      <mtu>1500</mtu>
      <txqueuelen>0</txqueuelen>
    </ip-info>
    <stats>
      <rx_packets>2971387</rx_packets>
      <rx_bytes>420208255</rx_bytes>
      <rx_errors>0</rx_errors>
      <rx_dropped>229</rx_dropped>
      <rx_overruns>0</rx_overruns>
      <rx_frame>0</rx_frame>
      <tx_packets>155</tx_packets>
      <tx_bytes>45522</tx_bytes>
      <tx_errors>0</tx_errors>
      <tx_dropped>0</tx_dropped>
      <tx_overruns>0</tx_overruns>
      <tx_carrier>0</tx_carrier>
      <tx_collisions>0</tx_collisions>
    </stats>
    <dhcp>
      <enabled>false</enabled>
      <offer>false</offer>
      <interface>NA</interface>
      <fixed_address>0.0.0.0</fixed_address>
      <subnet_mask>0.0.0.0</subnet_mask>
      <gateway>0.0.0.0</gateway>
      <lease_time>0</lease_time>
      <message_type>0</message_type>
      <name_servers>NA</name_servers>
      <server_identifier>0.0.0.0</server_identifier>
      <renewal_time>0</renewal_time>
      <rebinding_time>0</rebinding_time>
      <vendor_encapsulated_options>NA</vendor_encapsulated_options>
      <domain_name>NA</domain_name>
      <renew>0001-01-01T00:00:00-00:00</renew>
      <rebind>0001-01-01T00:00:00-00:00</rebind>
      <expire>0001-01-01T00:00:00-00:00</expire>
    </dhcp>
  </wan>
  <domain>NA</domain>
  <dns>
    <nameserver1>172.19.183.147</nameserver1>
    <nameserver2>0.0.0.0</nameserver2>
    <nameserver3>0.0.0.0</nameserver3>
  </dns>
 <hostname>Do3rdENCS75SettingsNoGW</hostname>
  <gateway>
    <ipv4_address>209.165.201.1</ipv4_address>
    <interface>MGMT</interface>
  </gateway>
</settings-native>
* Connection #0 to host 209.165.201.1 left intact
```

System Routes APIs

Table 3: System Routes APIs

Action	Method	Payload Required	API
To create a new route	POST	Yes	/api/config/system/routes
To modify an existing route	PUT	Yes	/api/config/system/routes/route/<host destination,netmask>
To retrieve the details of a route	GET	No	/api/operational/system/routes/route/<host destination,netmask> /api/config/system/routes
To delete a route	DELETE	No	/api/config/system/routes

Example for System Routes Payload

```
<route>
 <destination>209.165.201.1</destination>
 <prefixlen>16</prefixlen>
 <dev>lan-br</dev>
</route>
```

Table 4: System Routes Payload Description

Property	Type	Description	Mandatory/Default Value
destination	String	The route destination address.	Yes
prefixlen	Integer	The netmask for the destination address.	Yes
gateway	String	The gateway for the route.	No
dev	String	The device/interface that the route will use.	No

Note Though only the destination and prefixlen are mandatory parameters for creating a route, a valid route requires that you specify the gateway or the interface or both.

Example: POST System Route API

To create a new route:

```
curl -k -v -u "admin:admin" -H "Accept:application/vnd.yang.data+xml" -H
```

```
"Content-Type:application/vnd.yang.data+xml" -X POST
https://209.165.201.1/api/config/system/routes -d
"<route><destination>209.165.201.5</destination><prefixlen>16</prefixlen></route>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Aug 27 06:20:53 2016 GMT
*  expire date: Aug 25 06:20:53 2026 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> POST /api/config/system/routes HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.50.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 75
>
* upload completely sent off: 75 out of 75 bytes
< HTTP/1.1 201 Created
< Server: nginx/1.6.3
< Date: Sat, 27 Aug 2016 08:54:50 GMT
< Content-Type: text/html
< Content-Length: 0
< Location: https://209.165.201.1/api/config/system/routes/route/21.1.0.0,16
< Connection: keep-alive
< Last-Modified: Sat, 27 Aug 2016 08:54:49 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1472-288089-901692
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
```

Note The above example does not create a valid route because the gateway or device is not specified.

Example: PUT System Route API

```
curl -k -v -u "admin:admin" -H "Accept:application/vnd.yang.data+xml" -H
"Content-Type:application/vnd.yang.data+xml" -X PUT
```

```
https://209.165.201.1/api/config/system/routes/route/21.1.0.0,16 -d
"<route><destination>21.1.0.0</destination><prefixlen>16</prefixlen><dev>lan-br</dev></route>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Aug 27 06:20:53 2016 GMT
*  expire date: Aug 25 06:20:53 2026 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> PUT /api/config/system/routes/route/21.1.0.0,16 HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.50.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 92
>
* upload completely sent off: 92 out of 92 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.6.3
< Date: Sat, 27 Aug 2016 09:00:45 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Sat, 27 Aug 2016 09:00:45 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1472-288445-682999
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
```

Example: GET System Route API

To get route details and operational status for all routes:

```
curl -k -v -u "admin:admin" -X GET "https://209.165.201.1/api/operational/system/routes?deep"
Note: Unnecessary use of -X or --request, GET is already inferred.
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
```

```
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Aug 27 06:20:53 2016 GMT
*  expire date: Aug 25 06:20:53 2026 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/operational/system/routes?deep HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.50.1
> Accept: */*
>
< HTTP/1.1 200 OK
< Server: nginx/1.6.3
< Date: Sat, 27 Aug 2016 09:07:19 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<

<routes xmlns="http://www.cisco.com/nfv" xmlns:y="http://tail-f.com/ns/rest"
xmlns:system="http://www.cisco.com/nfv">
  <route>
    <destination>192.0.2.4</destination>
    <prefixlen>16</prefixlen>
    <gateway>192.0.2.1</gateway>
    <dev>lan-br</dev>
    <status>Success</status>
  </route>
  <route>
    <destination>192.0.2.5</destination>
    <prefixlen>16</prefixlen>
    <gateway>192.0.2.11</gateway>
    <dev>lan-br</dev>
    <status>Success</status>
  </route>
</routes>
* Connection #0 to host 209.165.201.1 left intact
```

Example: DELETE System Route API

```
curl -k -v -u "admin:admin" -H "Accept:application/vnd.yang.data+xml" -H
"Content-Type:application/vnd.yang.data+xml" -X DELETE
https://209.165.201.1/api/config/system/routes -d
"<route><destination>21.1.0.0</destination><prefixlen>16</prefixlen></route>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Aug 27 06:20:53 2016 GMT
*  expire date: Aug 25 06:20:53 2026 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> DELETE /api/config/system/routes HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46YWRtaW4=
> User-Agent: curl/7.50.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 75
>
* upload completely sent off: 75 out of 75 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.6.3
< Date: Sat, 27 Aug 2016 08:43:52 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Sat, 27 Aug 2016 08:43:52 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1472-287432-946952
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
```

VLAN APIs

The management VLAN is configured on the WAN interface.

Table 5: VLAN APIs

Action	Method	Payload Required	API
To configure a new VLAN or modify an existing VLAN	PUT	Yes	/api/config/bridges/bridge/wan-br/vlan

To get the configured VLAN info	GET	No	/api/config/bridges/bridge/wan2-br/vlan /api/config/bridges/bridge/user-br/vlan
To view the operational VLAN (the VLAN that is configured for the NFVIS management traffic on the wan-br).	GET	No	/api/operational/bridge-settings/bridge/wan-br/vlan
To delete a VLAN	DELETE	No	/api/config/bridges/bridge/wan-br/vlan

Example for VLAN Payload

```
<vlan> <vlan-id> </vlan>
```

The valid range for VLAN is from 1 to 4094.

Example: PUT VLAN API

Use the PUT VLAN API to create a new VLAN or modify an existing VLAN. When you modify a VLAN, the existing VLAN ID is replaced with the modified VLAN ID.

```
curl -k -v -u admin:Cisco#123 -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -k -X
PUT https://192.0.2.2/api/config/bridges/bridge/wan-br/vlan -d "<vlan>120</vlan>"
*   Trying 192.0.2.2...

* Connected to 192.0.2.2 (192.0.2.2) port 443 (#0)

* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH

* successfully set certificate verify locations:

*   CAfile: /etc/pki/tls/certs/ca-bundle.crt

  CApath: none

* TLSv1.0 (OUT), TLS handshake, Client hello (1):

* TLSv1.0 (IN), TLS handshake, Server hello (2):

* TLSv1.0 (IN), TLS handshake, Certificate (11):

* TLSv1.0 (IN), TLS handshake, Server key exchange (12):

* TLSv1.0 (IN), TLS handshake, Server finished (14):

* TLSv1.0 (OUT), TLS handshake, Client key exchange (16):

* TLSv1.0 (OUT), TLS change cipher, Client hello (1):

* TLSv1.0 (OUT), TLS handshake, Finished (20):

* TLSv1.0 (IN), TLS change cipher, Client hello (1):

* TLSv1.0 (IN), TLS handshake, Finished (20):
```

```
* SSL connection using TLSv1.0 / DHE-RSA-AES256-SHA
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Feb 15 23:33:39 2017 GMT
*  expire date: Feb 13 23:33:39 2027 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> PUT /api/config/system/settings/wan/vlan HTTP/1.1
> Host: 192.0.2.2
> Authorization: Basic YWRtaW46Q2lzY28jMTIz
> User-Agent: curl/7.49.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 16
>
* upload completely sent off: 16 out of 16 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.10.1
< Date: Thu, 16 Feb 2017 22:24:44 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Thu, 16 Feb 2017 22:24:36 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1487-283876-32584
< Pragma: no-cache
```

Example: GET VLAN API

Use this GET API to view the configured VLAN information.

```
curl -k -v -u admin:Cisco#123 -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/xml -k -X
```

```
GET https://192.0.2.2/api/config/bridges/bridge/wan-br/vlan
*   Trying 192.0.2.2...

* Connected to 192.0.2.2 (192.0.2.2) port 443 (#0)

* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH

* successfully set certificate verify locations:

*   CAfile: /etc/pki/tls/certs/ca-bundle.crt

  CApath: none

* TLSv1.0 (OUT), TLS handshake, Client hello (1):

* TLSv1.0 (IN), TLS handshake, Server hello (2):

* TLSv1.0 (IN), TLS handshake, Certificate (11):

* TLSv1.0 (IN), TLS handshake, Server key exchange (12):

* TLSv1.0 (IN), TLS handshake, Server finished (14):

* TLSv1.0 (OUT), TLS handshake, Client key exchange (16):

* TLSv1.0 (OUT), TLS change cipher, Client hello (1):

* TLSv1.0 (OUT), TLS handshake, Finished (20):

* TLSv1.0 (IN), TLS change cipher, Client hello (1):

* TLSv1.0 (IN), TLS handshake, Finished (20):

* SSL connection using TLSv1.0 / DHE-RSA-AES256-SHA

* Server certificate:

*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate

*  start date: Feb 15 23:33:39 2017 GMT

*  expire date: Feb 13 23:33:39 2027 GMT

*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate

*  SSL certificate verify result: self signed certificate (18), continuing anyway.

* Server auth using Basic with user 'admin'

> GET /api/config/system/settings/wan/vlan HTTP/1.1

> Host: 192.0.2.2

> Authorization: Basic YWRtaW46Q2lzY28jMTIz

> User-Agent: curl/7.49.1

> Accept:application/vnd.yang.data+xml

> Content-Type:application/xml
>
< HTTP/1.1 200 OK

< Server: nginx/1.10.1
< Date: Thu, 16 Feb 2017 22:43:21 GMT

< Content-Type: application/vnd.yang.data+xml

< Transfer-Encoding: chunked

< Connection: keep-alive

< Last-Modified: Thu, 16 Feb 2017 22:24:36 GMT

< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate

< Etag: 1487-283876-32584

< Pragma: no-cache
```

Use this GET API to view the operational VLAN (the VLAN that is configured for the NFVIS management traffic on the wan-br).

```
curl -k -v -u admin:Cisco#123 -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/xml -k -X
GET https://192.0.2.2/api/operational/bridge-settings/wan-br/vlan
*   Trying 192.0.2.2...

* Connected to 192.0.2.2 (192.0.2.2) port 443 (#0)

* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH

* successfully set certificate verify locations:

*   CAfile: /etc/pki/tls/certs/ca-bundle.crt

  CApath: none

* TLSv1.0 (OUT), TLS handshake, Client hello (1):

* TLSv1.0 (IN), TLS handshake, Server hello (2):

* TLSv1.0 (IN), TLS handshake, Certificate (11):

* TLSv1.0 (IN), TLS handshake, Server key exchange (12):

* TLSv1.0 (IN), TLS handshake, Server finished (14):

* TLSv1.0 (OUT), TLS handshake, Client key exchange (16):

* TLSv1.0 (OUT), TLS change cipher, Client hello (1):

* TLSv1.0 (OUT), TLS handshake, Finished (20):

* TLSv1.0 (IN), TLS change cipher, Client hello (1):

* TLSv1.0 (IN), TLS handshake, Finished (20):

* SSL connection using TLSv1.0 / DHE-RSA-AES256-SHA

* Server certificate:

*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate

*  start date: Feb 15 23:33:39 2017 GMT
*  expire date: Feb 13 23:33:39 2027 GMT

*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate

*  SSL certificate verify result: self signed certificate (18), continuing anyway.

* Server auth using Basic with user 'admin'

> GET /api/operational/system/settings-native/wan/vlan HTTP/1.1

> Host: 192.0.2.2

> Authorization: Basic YWRtaW46Q2lzY28jMTIz

> User-Agent: curl/7.49.1

> Accept:application/vnd.yang.data+xml

> Content-Type:application/xml

>

< HTTP/1.1 200 OK

< Server: nginx/1.10.1

< Date: Thu, 16 Feb 2017 22:44:37 GMT

< Content-Type: application/vnd.yang.data+xml

< Transfer-Encoding: chunked

< Connection: keep-alive

< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate

< Pragma: no-cache
<

<vlan xmlns="http://www.cisco.com/nfv" xmlns:y="http://tail-f.com/ns/rest"
xmlns:system="http://www.cisco.com/nfv">

 <tag>120</tag>
</vlan>
```

Example: DELETE VLAN API

```
curl -k -v -u admin:Cisco#123 -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -k -X
DELETE https://192.0.2.2/api/config/bridges/bridge/wan-br/vlan
*   Trying 192.0.2.2...

* Connected to 192.0.2.2 (192.0.2.2) port 443 (#0)

* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH

* successfully set certificate verify locations:

*   CAfile: /etc/pki/tls/certs/ca-bundle.crt

  CApath: none
```

```
* TLSv1.0 (OUT), TLS handshake, Client hello (1):
* TLSv1.0 (IN), TLS handshake, Server hello (2):
* TLSv1.0 (IN), TLS handshake, Certificate (11):
* TLSv1.0 (IN), TLS handshake, Server key exchange (12):
* TLSv1.0 (IN), TLS handshake, Server finished (14):
* TLSv1.0 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.0 (OUT), TLS change cipher, Client hello (1):
* TLSv1.0 (OUT), TLS handshake, Finished (20):
* TLSv1.0 (IN), TLS change cipher, Client hello (1):
* TLSv1.0 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.0 / DHE-RSA-AES256-SHA
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Feb 15 23:33:39 2017 GMT
*  expire date: Feb 13 23:33:39 2027 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> DELETE /api/config/system/settings/wan/vlan HTTP/1.1
> Host: 192.0.2.2
> Authorization: Basic YWRtaW46Q2lzY28jMTIz
> User-Agent: curl/7.49.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
>
< HTTP/1.1 204 No Content
< Server: nginx/1.10.1
< Date: Thu, 16 Feb 2017 22:48:59 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Thu, 16 Feb 2017 22:48:50 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate

< Etag: 1487-285330-811423

< Pragma: no-cache
```

User Management APIs

Action	Method	Payload Required	API
Add a user	POST	Yes	/api/config/rbac/authentication/users/create-user
Modify a user (Changing the user password)	POST	Yes	/api/operations/rbac/authentication/users /user/<user-name>/change-password
Change the user role	POST	Yes	/api/operations/rbac/authentication/users/user /oper/change-role
Get all users	GET	No	/api/config/rbac/authentication/users/user?deep
Delete a user	Delete	Yes	/api/config/rbac/authentication/users/delete-user
Configure the minimum password length	POST	Yes	/api/config/rbac/authentication/
Configure the password lifetime	POST	Yes	/api/config/rbac/authentication/password-lifetime/
Configure the account inactivity period	POST	Yes	/api/config/rbac/authentication/account-inactivity/
Activate an inactive user account	POST	No	/api/operations/rbac/authentication/users/user/username/activate

Example for Add User Payload

```
<input>
 <name>testuser</name>
 <password>Test123#</password>
 <role>operators</role>
</input>
```

Example for Change Role Payload

```
<input>
 <old-role>auditors</old-role>
 <new-role>operators</new-role>
</input>
```

Example for Change Password Payload

```
<input>
 <old-password>Hello123#</old-password>
 <new-password>Hello123$</new-password>
 <confirm-password>Hello123$</confirm-password>
</input>
```

Example for Minimum Password Length Payload

```
<min-pwd-length>9</min-pwd-length>
```

Example for Password Lifetime Payload

```
<enforce>true</enforce>
<min-days>7</min-days>
<max-days>30</max-days>
```

Example for Account Inactivity Period Payload

```
<enforce>true</enforce>
<inactivity-days>50</inactivity-days>
```

Table 6: User Management API Payload Description

Property	Type	Description	Mandatory/Default Value
name	String	Name of the user	No
role	String	Role of the user	Yes
password	String	Password of the user	Yes
old-role	String	Existing role of the user	Yes
new-role	String	New role of the user	Yes
old-password	String	Existing password	Yes
new-password	String	New password for the user	Yes
confirm-password	String	Confirms the new password	Yes
min-pwd-length	Number	Minimum length required for passwords of all users. The minimum length must be between 7 to 128 characters.	Yes
enforce	String	Enforces or removes the rule. Valid values for this parameter are true and false.	Yes
min-days	Number	Number of days after which the users can change the password.	Yes
max-days	Number	Number of days before which the users must change the password.	Yes

inactivity-days	Number	Number of days after which an unused account is marked as inactive.	Yes

Example: POST Add User API

```
curl -X POST -v -k -u admin:Admin123$
https://209.165.201.1/api/operations/rbac/authentication/users/create-user -H
Content-Type:application/vnd.yang.data+xml
-d"<input><name>testname</name><password>Hello123#</password><role>operators</role></input>"
```

Example: POST Change Role API

```
curl -X POST -v -k -u admin:Cisco123#
https://209.165.201.1/api/operations/rbac/authentication/users/user/oper/change-role
 -H Content-Type:application/vnd.yang.data+xml -d
"<input><old-role>auditors</old-role><new-role>operators</new-role></input>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate
* Server auth using Basic with user 'admin'
> POST /api/operations/rbac/authentication/users/user/oper/change-role HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 74
>
* upload completely sent off: 74 out of 74 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.10.1
< Date: Thu, 16 Feb 2017 20:51:03 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
```

Example: POST Change Password API

```
curl -X POST -v -k -u admin:Admin123#
https://209.165.201.1/api/operations/rbac/authentication/users/user/testuser12/change-password
 -H
 Content-Type:application/vnd.yang.data+xml -d
"<input><old-password>Hello123#</old-password><new-password>Hello123$</new-password>
<confirm-password>Hello123$</confirm-password></input>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate
* Server auth using Basic with user 'admin'
> POST /api/operations/rbac/authentication/users/user/testuser12/change-password HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 137
>
* upload completely sent off: 137 out of 137 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.6.3
< Date: Thu, 22 Dec 2016 19:05:10 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
```

Example: GET Users API

```
curl -X GET -v -k -u "admin:Admin123#" -H "Content-Type: application/vnd.yang.collection+xml"
 -H "Accept: application/vnd.yang.collection+xml"
 "https://209.165.201.1/api/config/rbac/authentication/users/user?deep"
<collection xmlns:y="http://tail-f.com/ns/rest">
  <user xmlns="http://www.cisco.com/nfv/rbac">
    <name>admin</name>
    <role>administrators</role>
    <password>$7$K1dMMts4XHgdT//+YGlrrqh4YCZvYye4</password>
    <y:operations>

<change-password>/api/config/rbac/authentication/users/user/admin/_operations/change-password</change-password>


<change-role>/api/config/rbac/authentication/users/user/admin/_operations/change-role</change-role>

    </y:operations>
  </user>
  <user xmlns="http://www.cisco.com/nfv/rbac">
    <name>oper</name>
    <role>administrators</role>
    <password>$7$u76ZWuWU1Kn+gCPsImgEKpBkavgziDuO</password>
    <y:operations>

<change-password>/api/config/rbac/authentication/users/user/oper/_operations/change-password</change-password>


<change-role>/api/config/rbac/authentication/users/user/oper/_operations/change-role</change-role>

    </y:operations>
  </user>
  <user xmlns="http://www.cisco.com/nfv/rbac">
    <name>testuser12</name>
    <role>administrators</role>
    <password>$7$YhK1LGI2HTjzCTBVDZ8lxfWxTvqjjcvN</password>
    <y:operations>

<change-password>/api/config/rbac/authentication/users/user/testuser12/_operations/change-password</change-password>


<change-role>/api/config/rbac/authentication/users/user/testuser12/_operations/change-role</change-role>
```

```
    </y:operations>
  </user>
</collection>
```

Example: Delete User API

```
curl -X POST -v -k -u admin:Admin123#
https://209.165.201.1/api/operations/rbac/authentication/users/delete-user -H
Content-Type:application/vnd.yang.data+xml -d"<input><name>testname</name></input>"
```

Example: POST Configure Minimum Password Length

```
curl -X POST -v -k -u admin:Admin123# https://209.165.201.1/api/config/rbac/authentication/
 -H Content-Type:application/vnd.yang.data+xml -d "<min-pwd-length>9</min-pwd-length>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: nfvis
* Server auth using Basic with user 'admin'
> POST /api/config/rbac/authentication/ HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46QWRtaW4jMTIz
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 34
>
* upload completely sent off: 34 out of 34 bytes
< HTTP/1.1 204 No Content
< Server: nginx
< Date: Tue, 31 Oct 2017 11:56:36 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
```

Examples: POST Configure Password Lifetime

```
curl -X POST -v -k -u admin:Admin#123
https://209.165.201.1/api/config/rbac/authentication/password-lifetime/ -H
Content-Type:application/vnd.yang.data+xml -d "<enforce>true</enforce>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: nfvis
* Server auth using Basic with user 'admin'
> POST /api/config/rbac/authentication/password-lifetime/ HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46QWRtaW4jMTIz
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 23
```

```
>
* upload completely sent off: 23 out of 23 bytes
< HTTP/1.1 204 No Content
< Server: nginx
< Date: Tue, 31 Oct 2017 11:59:48 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact

curl -X POST -v -k -u admin:Admin#123
https://209.165.201.1/api/config/rbac/authentication/password-lifetime/ -H
Content-Type:application/vnd.yang.data+xml -d "<min-days>1</min-days>"

*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: nfvis
* Server auth using Basic with user 'admin'
> POST /api/config/rbac/authentication/password-lifetime/ HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46QWRtaW4jMTIz
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 23
>
* upload completely sent off: 23 out of 23 bytes
< HTTP/1.1 204 No Content
< Server: nginx
< Date: Tue, 31 Oct 2017 11:59:48 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact

curl -X POST -v -k -u admin:Admin#123
https://209.165.201.1/api/config/rbac/authentication/password-lifetime/ -H
Content-Type:application/vnd.yang.data+xml -d "<max-days>30</max-days>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: nfvis
* Server auth using Basic with user 'admin'
> POST /api/config/rbac/authentication/password-lifetime/ HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46QWRtaW4jMTIz
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 23
>
* upload completely sent off: 23 out of 23 bytes
< HTTP/1.1 204 No Content
< Server: nginx
< Date: Tue, 31 Oct 2017 11:59:48 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
```

Examples: POST Configure Account Inactivity Period

```
curl -X POST -v -k -u admin:Admin#123
https://209.165.201.1/api/config/rbac/authentication/account-inactivity/ -H
Content-Type:application/vnd.yang.data+xml -d "<enforce>true</enforce>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: nfvis
* Server auth using Basic with user 'admin'
> POST /api/config/rbac/authentication/account-inactivity/ HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46QWRtaW4jMTIz
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 23
>
* upload completely sent off: 23 out of 23 bytes
< HTTP/1.1 204 No Content
< Server: nginx
< Date: Tue, 31 Oct 2017 12:00:52 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
```

```
curl -X POST -v -k -u admin:Admin#123
https://209.165.201.1/api/config/rbac/authentication/account-inactivity/ -H
Content-Type:application/vnd.yang.data+xml -d "<inactivity-days>50</inactivity-days>"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: nfvis
* Server auth using Basic with user 'admin'
> POST /api/config/rbac/authentication/account-inactivity/ HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46QWRtaW4jMTIz
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 23
>
* upload completely sent off: 23 out of 23 bytes
< HTTP/1.1 204 No Content
< Server: nginx
< Date: Tue, 31 Oct 2017 12:00:52 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
```

Example: POST Activate an Inactive User Account

```
curl -X POST -v -k -u admin:Admin#123
https://209.165.201.1/api/operations/rbac/authentication/users/user/guest_user/activate -H
 Con-tent-Type:application/vnd.yang.data+xml
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: nfvis
* Server auth using Basic with user 'admin'
> POST /api/operations/rbac/authentication/users/user/guest_user/activate HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46QWRtaW4jMTIz
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
>
< HTTP/1.1 204 No Content
< Server: nginx
< Date: Tue, 31 Oct 2017 12:11:31 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
* Connection #0 to host 209.165.201.1 left intact
```

TACACS+ Server APIs

Table 7: TACACS+ Server APIs

Action	Method	Payload Required	API
To configure a TACACS+ server	POST	Yes	/api/config/security_servers/tacacs-server
To modify a TACACS+ server configuration	PUT	Yes	/api/config/security_servers/tacacs-server
To get the TACACS+ server configuration details	GET	No	/api/config/security_servers/tacacs-server?deep
To delete a TACACS+ server configuration	DELETE	No	/api/config/security_servers/tacacs-server /host/<ip-address/domain-name>

Example for TACACS+ Server Payload

Table 8: TACACS+ Server Payload Description

Property	Type	Description	Mandatory/Default Value

Example: POST TACACS Server API

```
curl -k -v -u "admin:cisco123" -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+json -X
POST https://172.19.181.173/api/config/security_servers/tacacs-server -d
* Hostname was NOT found in DNS cache
*   Trying 172.19.181.173...
* Connected to 172.19.181.173 (172.19.181.173) port 443 (#0)
* successfully set certificate verify locations:
*   CAfile: none
  CApath: /etc/ssl/certs
* SSLv3, TLS handshake, Client hello (1):
* SSLv3, TLS handshake, Server hello (2):
* SSLv3, TLS handshake, CERT (11):
* SSLv3, TLS handshake, Server key exchange (12):
* SSLv3, TLS handshake, Server finished (14):
* SSLv3, TLS handshake, Client key exchange (16):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSL connection using ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*        subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*        start date: 2017-01-13 23:47:41 GMT
*        expire date: 2027-01-11 23:47:41 GMT
*        issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*        SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> POST /api/config/security_servers/tacacs-server HTTP/1.1
> Authorization: Basic YWRtaW46Y2lzY28xMjM=
> User-Agent: curl/7.35.0
> Host: 172.19.181.173
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+json
> Content-Length: 122
>
* upload completely sent off: 122 out of 122 bytes
< HTTP/1.1 201 Created
* Server nginx/1.10.1 is not blacklisted
< Server: nginx/1.10.1
< Date: Mon, 27 Feb 2017 18:14:46 GMT
< Content-Type: text/html
< Content-Length: 0
< Location: https://172.19.181.173/api/config/security_servers/tacacs-server/host/5.5.5.5
< Connection: keep-alive
< Last-Modified: Mon, 27 Feb 2017 18:14:46 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1488-219286-189602
< Pragma: no-cache
```

Example: GET TACACS Server API

```
curl -k -v -u "admin:cisco123" -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+json -X
GET https://209.165.201.1/api/config/security_servers/tacacs-server?deep
* Hostname was NOT found in DNS cache
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* successfully set certificate verify locations:
```

```
*   CAfile: none
  CApath: /etc/ssl/certs
* SSLv3, TLS handshake, Client hello (1):
* SSLv3, TLS handshake, Server hello (2):
* SSLv3, TLS handshake, CERT (11):
* SSLv3, TLS handshake, Server key exchange (12):
* SSLv3, TLS handshake, Server finished (14):
* SSLv3, TLS handshake, Client key exchange (16):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSL connection using ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*        subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*        start date: 2017-01-13 23:47:41 GMT
*        expire date: 2027-01-11 23:47:41 GMT
*        issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*        SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/config/security_servers/tacacs-server?deep HTTP/1.1
> Authorization: Basic YWRtaW46Y2lzY28xMjM=
> User-Agent: curl/7.35.0
> Host: 209.165.201.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+json
>
< HTTP/1.1 200 OK
* Server nginx/1.10.1 is not blacklisted
< Server: nginx/1.10.1
< Date: Mon, 27 Feb 2017 18:07:49 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Last-Modified: Fri, 24 Feb 2017 01:13:51 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1487-898831-958028
< Pragma: no-cache
<

<tacacs-server xmlns="http://www.cisco.com/ns/test/security" xmlns:y="http://tail-

f.com/ns/rest"  xmlns:security="http://www.cisco.com/ns/test/security">
  <host>
    <server>10.2.2.2</server>
    <secret>
      <key>0</key>
      <shared-secret>tac22</shared-secret>
    </secret>
  </host>
  <host>
    <server>10.3.3.3</server>
    <secret>
      <key>0</key>
      <shared-secret>tac22</shared-secret>
    </secret>
  </host>
  <host>
    <server>10.1.1.1</server>
    <secret>
      <key>0</key>
      <shared-secret>tac22</shared-secret>
    </secret>
  </host>
```

```
</tacacs-server>
```

Example: PUT TACACS Server API

```
* Hostname was NOT found in DNS cache
*   Trying 172.19.181.173...
* Connected to 172.19.181.173 (172.19.181.173) port 443 (#0)
* successfully set certificate verify locations:
*   CAfile: none
  CApath: /etc/ssl/certs
* SSLv3, TLS handshake, Client hello (1):
* SSLv3, TLS handshake, Server hello (2):
* SSLv3, TLS handshake, CERT (11):
* SSLv3, TLS handshake, Server key exchange (12):
* SSLv3, TLS handshake, Server finished (14):
* SSLv3, TLS handshake, Client key exchange (16):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSL connection using ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*        subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*        start date: 2017-01-13 23:47:41 GMT
*        expire date: 2027-01-11 23:47:41 GMT
*        issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*        SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> PUT /api/config/security_servers/tacacs-server/host/5.5.5.5 HTTP/1.1
> Authorization: Basic YWRtaW46Y2lzY28xMjM=
> User-Agent: curl/7.35.0
> Host: 172.19.181.173
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+json
> Content-Length: 92
>
* upload completely sent off: 92 out of 92 bytes
< HTTP/1.1 204 No Content
* Server nginx/1.10.1 is not blacklisted
< Server: nginx/1.10.1
< Date: Mon, 27 Feb 2017 18:20:13 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Mon, 27 Feb 2017 18:20:13 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1488-219613-571277
< Pragma: no-cache
```

Example: DELETE TACACS Server API

```
curl -k -v -u "admin:cisco123" -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+json -X
DELETE https://192.0.2.2/api/config/security_servers/tacacs-server/host/5.5.5.5
* Hostname was NOT found in DNS cache
*   Trying 192.0.2.2...
* Connected to 192.0.2.2 (192.0.2.2) port 443 (#0)
* successfully set certificate verify locations:
```

```
*   CAfile: none
  CApath: /etc/ssl/certs
* SSLv3, TLS handshake, Client hello (1):
* SSLv3, TLS handshake, Server hello (2):
* SSLv3, TLS handshake, CERT (11):
* SSLv3, TLS handshake, Server key exchange (12):
* SSLv3, TLS handshake, Server finished (14):
* SSLv3, TLS handshake, Client key exchange (16):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSLv3, TLS change cipher, Client hello (1):
* SSLv3, TLS handshake, Finished (20):
* SSL connection using ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*        subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*        start date: 2017-01-13 23:47:41 GMT
*        expire date: 2027-01-11 23:47:41 GMT
*        issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*        SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> DELETE /api/config/security_servers/tacacs-server/host/5.5.5.5 HTTP/1.1
> Authorization: Basic YWRtaW46Y2lzY28xMjM=
> User-Agent: curl/7.35.0
> Host: 192.0.2.2
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+json
>
< HTTP/1.1 204 No Content
* Server nginx/1.10.1 is not blacklisted
< Server: nginx/1.10.1
< Date: Mon, 27 Feb 2017 18:21:30 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Mon, 27 Feb 2017 18:21:30 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1488-219690-404414
< Pragma: no-cache
```

Trusted IP Connection APIs

Table 9: Trusted IP Connection APIs

Action	Method	Payload Required	API
To add, modify, or remove the trusted source IP connection	PUT	Yes	/api/config/system/settings
To verify the configuration of the trusted source IP addresses	GET	No	/api/operational/system/settings-native/trusted-source
To verify the system settings	GET	No	/api/operational/system/settings-native?deep
To verify the trusted source or system settings	GET	No	/api/operational/system/settings?deep

Example for the Trusted IP Connection Payload

```
<settings>
    <hostname>nfvis</hostname>
    <trusted-source>192.0.2.0/24</trusted-source>
        <mgmt>
            <ip>
                <address>198.51.100.1</address>
                <netmask>255.255.255.0</netmask>
            </ip>
        </mgmt>
        <wan>
            <ip>
                 <address>198.51.100.2</address>
                 <netmask>255.255.255.0</netmask>
            </ip>
        </wan>
    <default-gw>198.51.100.3</default-gw>
</settings>
```

Table 10: Trusted IP Connection Payload Description

Property	Type	Description	Mandatory/Default Value
hostname	String	Hostname of the system	Yes
trusted-source	String	Source IP address You can specify a single IP address or a range of IP addresses.	No
mgmt ip address netmask	String	Specifies the management IP address and netmask.	Yes
wan ip address netmask	String	Specifies the WAN IP address and netmask.	Yes
default-gw	String	IP address of the default gateway	Yes

Example: PUT Trusted IP Connection API

Use this API to add, modify, or remove the trusted source IP address or addresses.

Note To delete all trusted source IP addresses, you need to remove the trusted source element (trusted-source) from the payload. You can modify a trusted source IP address by replacing it with a new IP address.

```
curl -k -v -u "admin:Cisco123#" -H "Content-Type:application/vnd.yang.data+xml" -X PUT
https://198.51.100.1/api/config/system/settings
-d "<settings><hostname>nfvis</hostname><trusted-source>192.0.2.0/24</trusted-source>
<mgmt><ip><address>198.51.100.1</address><netmask>255.255.255.0</netmask></ip></mgmt>
<wan><ip><address>198.51.100.2</address><netmask>255.255.255.0</netmask></ip></wan><default-gw>198.51.100.3</default-gw></settings>"

*   Trying 198.51.100.1...
* Connected to 198.51.100.1 (198.51.100.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Mar 14 06:53:22 2017 GMT
*  expire date: Mar 12 06:53:22 2027 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> PUT /api/config/system/settings HTTP/1.1
> Host: 198.51.100.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.50.1
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 343
>
* upload completely sent off: 343 out of 343 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.10.1
< Date: Tue, 14 Mar 2017 21:19:21 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Tue, 14 Mar 2017 21:19:15 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1489-526355-690730
< Pragma: no-cache
<
* Connection #0 to host 198.51.100.1 left intact
```

Example: GET Trusted IP Connection API

```
curl -v -k -u  admin:Cisco123# -X GET
'https://198.51.100.1/api/operational/system/settings-native/trusted-source'

Note: Unnecessary use of -X or --request, GET is already inferred.
*   Trying 198.51.100.1...
* Connected to 198.51.100.1 (198.51.100.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
```

```
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Mar 14 06:53:22 2017 GMT
*  expire date: Mar 12 06:53:22 2027 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/operational/system/settings-native/trusted-source HTTP/1.1
> Host: 198.51.100.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.50.1
> Accept: */*
>
< HTTP/1.1 200 OK
< Server: nginx/1.10.1
< Date: Tue, 14 Mar 2017 21:08:49 GMT
< Content-Type: application/vnd.yang.collection+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
<collection xmlns:y="http://tail-f.com/ns/rest">
  <trusted-source xmlns="http://www.cisco.com/nfv">192.0.2.0/24</trusted-source>
  </collection>
* Connection #0 to host 198.51.100.1 left intact
```

Banner and Message APIs

Table 11: Banner and Message APIs

Action	Method	Payload Required	API
To configure or update a banner or message of the day or both	PUT	Yes	/api/config/banner-motd
To get system banner details and user-defined banner and message of the day	GET	No	/api/operational/banner-motd /api/operational/banner-motd/system-banner /api/operational/banner-motd/banner /api/operational/banner-motd/motd

To get user-defined banner and message of the day details	GET	No	/api/config/banner-motd /api/config/banner-motd/banner /api/config/banner-motd/motd
To delete the user-defined banner or message of the day	DELETE	No	/api/config/banner-motd /api/config/banner-motd/banner /api/config/banner-motd/motd

Example for Banner and Message Payload

```
<banner-motd>
    <banner> my banner </banner>
    <motd> my motd </motd>
</banner-motd>
```

Table 12: Banner and Message Payload Description

Property	Type	Description	Mandatory/Default Value
banner	String	Specifies the user-defined banner.	No
motd	String	Message of the day	No

Example: PUT Banner-MOTD API

```
curl -k -v -u "admin:Cisco123*" -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
PUT https://209.165.201.1/api/config/banner-motd -d '<banner-motd><banner>my
banner</banner><motd>my motd</motd></banner-motd>'
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate
* Server auth using Basic with user 'admin'
> PUT /api/config/banner-motd HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMq
> User-Agent: curl/7.43.0
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 99
>
* upload completely sent off: 99 out of 99 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.6.3
< Date: Tue, 27 Dec 2016 01:48:31 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Tue, 27 Dec 2016 01:48:31 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1482-803311-573328
< Pragma: no-cache
```

Example: GET Banner-MOTD API

Use this operational API to get information about the system-defined banner.

```
curl -k -v -u "admin:Cisco123*" -X GET
"https://209.165.201.1/api/operational/banner-motd/system-banner"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate
* Server auth using Basic with user 'admin'
> GET /api/operational/banner-motd HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMq
> User-Agent: curl/7.43.0
> Accept: */*
>
< HTTP/1.1 200 OK
< Server: nginx/1.6.3
< Date: Tue, 27 Dec 2016 01:50:24 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<

<banner-motd xmlns="http://www.cisco.com/nfvis/banner" xmlns:y="http://tail-f.com/ns/rest"
  xmlns:banner_motd="http://www.cisco.com/nfvis/banner">
  <banner>---my banner 111
2222
3333</banner>
  <motd>----my motd 1111</motd>
  <system-banner>
Cisco Enterprise Network Function Virtualization Infrastructure Software (NFVIS)

Copyright (c) 2015-2016 by Cisco Systems, Inc.
Cisco, Cisco Systems, and Cisco Systems logo are registered trademarks of Cisco
Systems, Inc. and/or its affiliates in the U.S. and certain other countries.

The copyrights to certain works contained in this software are owned by other
third parties and used and distributed under third party license agreements.
Certain components of this software are licensed under the GNU GPL 2.0, GPL 3.0,
LGPL 2.1, LGPL 3.0 and AGPL 3.0.


</system-banner>
</banner-motd>
```

Use this GET API to get information about the user-defined banner and message of the day.

```
curl -k -v -u "admin:Cisco123*" -X GET "https://209.165.201.1/api/config/banner-motd"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate
* Server auth using Basic with user 'admin'
> GET /api/config/banner-motd HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMq
> User-Agent: curl/7.43.0
> Accept: */*
>
< HTTP/1.1 200 OK
< Server: nginx/1.6.3
< Date: Tue, 27 Dec 2016 01:51:58 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Last-Modified: Tue, 27 Dec 2016 01:48:31 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1482-803311-573328
< Pragma: no-cache
<

<banner-motd xmlns="http://www.cisco.com/nfvis/banner" xmlns:y="http://tail-f.com/ns/rest"
  xmlns:banner_motd="http://www.cisco.com/nfvis/banner">
  <banner>my banner</banner>
  <motd>my motd</motd>
</banner-motd>
```

Example: DELETE Banner-MOTD API

Use this DELETE API to delete the user-defined banner.

```
curl -k -v -u "admin:Cisco123*" -X DELETE
"https://209.165.201.1/api/config/banner-motd/banner"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate
* Server auth using Basic with user 'admin'
> DELETE /api/config/banner-motd/banner HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMq
> User-Agent: curl/7.43.0
> Accept: */*
>
< HTTP/1.1 204 No Content
< Server: nginx/1.6.3
< Date: Wed, 08 Feb 2017 20:27:29 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Wed, 08 Feb 2017 20:27:29 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1486-585649-542089
< Pragma: no-cache
```

Use this DELETE API to delete the user-defined message of the day.

```
curl -k -v -u "admin:Cisco123*" -X DELETE "https://209.165.201.1/api/config/banner-motd/motd"
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate
* Server auth using Basic with user 'admin'
> DELETE /api/config/banner-motd/motd HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMq
> User-Agent: curl/7.43.0
> Accept: */*
>
< HTTP/1.1 204 No Content
< Server: nginx/1.6.3
< Date: Wed, 08 Feb 2017 20:33:52 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Wed, 08 Feb 2017 20:33:52 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1486-586032-109043
< Pragma: no-cache
<
```

Note After deleting the banner or message of the day, you can run the GET operational API to confirm the deletion. If you use the parameter "banner" or "motd" along with the GET API, you get a 404 error if the deletion is successful. If you run the GET API without the parameter (/api/operational/banner-motd), you get the output with empty "banner-motd" tag, if the deletion is successful.

Disk Space APIs

Table 13: Disk Space API

Action	Method	Payload Required	API
To get the information on disk space	GET	Yes	/api/operational/system/disk-space

Example: GET Disk Space API

```
curl -k -v -u "admin:admin" -X GET
"https://209.165.201.1/api/operational/system/disk-space?deep"
Note: Unnecessary use of -X or --request, GET is already inferred.
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
```

```
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=nfvis
*  start date: Oct 23 17:25:04 2018 GMT
*  expire date: Oct 22 17:25:04 2023 GMT
*  issuer: CN=nfvis
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/operational/system/disk-space?deep HTTP/1.1
> Host: 172.25.221.106
> Authorization: Basic YWRtaW46MTIzI0FkbWlu
> User-Agent: curl/7.50.1
> Accept: */*
>
< HTTP/1.1 200 OK
< Server: nginx
< Date: Fri, 26 Oct 2018 01:10:37 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
< Content-Security-Policy: default-src https: 'unsafe-eval' 'unsafe-inline';img-src 'self'
 data:; object-src 'none'; connect-src 'self' *
< X-Frame-Options: SAMEORIGIN
< Strict-Transport-Security: max-age=31536000; includeSubDomains
< Cache-Control: max-age=0, no-cache, no-store, must-revalidate
<

<disk-space xmlns="http://www.cisco.com/nfv" xmlns:y="http://tail-f.com/ns/rest"
xmlns:system="http://www.cisco.com/nfv">
  <disk-info>
    <disk-name>lv_data</disk-name>
    <associated-physical-disk>sde2</associated-physical-disk>
    <total-size>41G</total-size>
    <size-used>8.6G</size-used>
    <size-available>32G</size-available>
    <use-percent>22%</use-percent>
  </disk-info>
  <disk-info>
    <disk-name>lv_var</disk-name>
    <associated-physical-disk>sde2</associated-physical-disk>
    <total-size>2.0G</total-size>
    <size-used>118M</size-used>
    <size-available>1.7G</size-available>
    <use-percent>7%</use-percent>
  </disk-info>
  <disk-info>
    <disk-name>lv_root</disk-name>
    <associated-physical-disk>sde2</associated-physical-disk>
    <total-size>7.8G</total-size>
    <size-used>1.8G</size-used>
    <size-available>5.7G</size-available>
    <use-percent>24%</use-percent>
  </disk-info>
  <disk-info>
    <disk-name>extdatastore2</disk-name>
    <associated-physical-disk>sdd</associated-physical-disk>
    <total-size>1.8T</total-size>
    <size-used>77M</size-used>
    <size-available>1.7T</size-available>
    <use-percent>1%</use-percent>
  </disk-info>
</disk-space>
* Connection #0 to host 209.165.201.1 left intact
```

System Time APIs

Table 14: System Time APIs

Action	Method	Payload Required	API
To set the manual time	PUT	Yes	• /api/config/system/time/set-manual-time
To configure the preferred and backup servers	PUT	Yes	• /api/config/system/time/ntp/preferred_server • /api/config/system/time/ntp/backup_server
To set the timezone	PUT	Yes	/api/config/system/time/timezone
To get the system time information	GET	No	/api/operational/system/time
To add NTP IPv6 server	POST	Yes	/api/config/system/time/
To delete NTP IPv6 server	DELETE	No	/api/config/system/time/ntp-ipv6/
To get time status	GET	NO	/api/operational/system/time

Example for System Time API Payload

```
<input><time>2017-01-01T00:00:00</time></input>
<preferred_server><ip-address></preferred_server>
<backup_server><ip-address></backup_server>
<timezone><zone/subzone></timezone>
<ntp-ipv6><ntp-server>2001:420:30d:201:ffff:ffff:fff4:35</ntp-server></ntp-ipv6>
```

Table 15: System Time API Payload Description

Property	Type	Description	Mandatory/Default Value

set-manual-time	String	Specifies manual time in YYYY-MM-DDTHH:MM:SS format.	Yes
preferred_server	String	Preferred server IP address or domain name.	Yes
backup_server	String	Backup server IP address or domain name.	No
timezone	String	Specifies the timezone.	No
ntp-server	String	Specifes the IPv6 address or domain name.	Yes

Example: PUT System Time Manual Time API

```
curl -v -k -u admin:Cisco123* -H "Content-Type: application/vnd.yang.data+xml" -X
PUT https://209.165.201.1/api/config/system/time/set-manual-time -d
'<input><time>2017-01-01T00:00:00</time></input>’

*   Trying 209.165.201.1...

* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)

* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate

* Server auth using Basic with user 'admin'

> PUT /api/config/system/time/set-manual-time HTTP/1.1

> Host: 209.165.201.1

> Authorization: Basic YWRtaW46Q2lzY28xMjMq

> User-Agent: curl/7.43.0

> Accept: */*

> Content-Type:application/vnd.yang.data+xml

> Content-Length: 46

>
* upload completely sent off: 46 out of 46 bytes

< HTTP/1.1 204 No Content

< Server: nginx/1.6.3

< Date: Wed, 01 Jan 2020 11:11:51 GMT

< Content-Type: text/html

< Content-Length: 0

< Connection: keep-alive

< Last-Modified: Wed, 30 Nov 2016 04:10:28 GMT

< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1480-479028-836845

< Pragma: no-cache

<
```

Example: PUT System Time Preferred Server API

```
curl -v -k -u admin:Cisco123* -H "Content-Type: application/vnd.yang.data+xml" -X
PUT https://209.165.201.1/api/config/system/time/ntp/preferred_server -d
'<preferred_server>209.165.201.2</preferred_server>'

*   Trying 209.165.201.1...

* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)

* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate

* Server auth using Basic with user 'admin'

> PUT /api/config/system/time/ntp/preferred_server HTTP/1.1

> Host: 209.165.201.1

> Authorization: Basic YWRtaW46Q2lzY28xMjMq

> User-Agent: curl/7.43.0

> Accept: */*

> Content-Type: application/vnd.yang.data+xml

> Content-Length: 49

>

* upload completely sent off: 49 out of 49 bytes

< HTTP/1.1 204 No Content

< Server: nginx/1.6.3

< Date: Wed, 01 Jan 2020 11:15:02 GMT

< Content-Type: text/html

< Content-Length: 0

< Connection: keep-alive

< Last-Modified: Wed, 01 Jan 2020 11:15:02 GMT

< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate

< Etag: 1480-479262-370866

< Pragma: no-cache
```

Example: PUT System Time Backup Server API

```
curl -v -k -u admin:Cisco123* -H "Content-Type: application/vnd.yang.data+xml" -X
PUT https:// 209.165.201.1/api/config/system/time/ntp/backup_server -d
'<backup_server>209.165.201.4</backup_server>’

  Trying 209.165.201.1...

* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)

* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate

* Server auth using Basic with user 'admin'

> PUT /api/config/system/time/ntp/backup_server HTTP/1.1

> Host: 209.165.201.1

> Authorization: Basic YWRtaW46Q2lzY28xMjMq

> User-Agent: curl/7.43.0

> Accept: */*

> Content-Type: application/vnd.yang.data+xml

> Content-Length: 43

>

* upload completely sent off: 43 out of 43 bytes

< HTTP/1.1 204 No Content

< Server: nginx/1.6.3

< Date: Wed, 01 Jan 2020 11:16:47 GMT

< Content-Type: text/html

< Content-Length: 0

< Connection: keep-alive

< Last-Modified: Wed, 01 Jan 2020 11:16:47 GMT

< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate

< Etag: 1480-479368-378871

< Pragma: no-cache
```

Example: PUT System Time Timezone API

```
curl -v -k -u admin:Cisco123* -H "Content-Type: application/vnd.yang.data+xml" -X
PUT https://209.165.201.1/api/config/system/time/timezone -d
'<timezone>America/New_York</timezone>'
```

```
*   Trying 209.165.201.1...

* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)

* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384

* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate

* Server auth using Basic with user 'admin'

> PUT /api/config/system/time/timezone HTTP/1.1

> Host: 209.165.201.1

> Authorization: Basic YWRtaW46Q2lzY28xMjMq

> User-Agent: curl/7.43.0

> Accept: */*

> Content-Type: application/vnd.yang.data+xml

> Content-Length: 37

>
* upload completely sent off: 37 out of 37 bytes

< HTTP/1.1 204 No Content

< Server: nginx/1.6.3

< Date: Wed, 01 Jan 2020 11:19:44 GMT

< Content-Type: text/html

< Content-Length: 0

< Connection: keep-alive

< Last-Modified: Wed, 01 Jan 2020 16:19:44 GMT

< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate

< Etag: 1480-479547-383761

< Pragma: no-cache
```

Example: GET System Time API

```
curl -v -k -u admin:Cisco123* -H "Content-Type: application/vnd.yang.data+xml" -X
GET https://209.165.201.1/api/operational/system/time?deep

*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* TLS 1.2 connection using TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
* Server certificate: Cisco-Enterprise-NFVIS-Self-Signed-Certificate
* Server auth using Basic with user 'admin'
> GET /api/operational/system/host_time HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMq
> User-Agent: curl/7.43.0
> Accept: */*
> Content-Type: application/vnd.yang.data+xml
>
< HTTP/1.1 200 OK
< Server: nginx/1.6.3
< Date: Wed, 01 Jan 2020 11:21:13 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
<time xmlns="http://www.cisco.com/nfv" xmlns:y="http://tail-f.com/ns/rest"
xmlns:system="http://www.cisco.com/nfv">
<ntp>
<status>
<remote>209.165.201.4</remote>
<refid>.GPS.</refid>
<st>1</st>
<t>u</t>
<when>2</when>
<poll>512</poll>
<reach>377</reach>
<delay>71.547</delay>
<offset>-1.862</offset>
<jitter>0.764</jitter>
</status>
</ntp>
<current-time>2017-01-01T12:12:12</current-time>
<current-timezone>UTC (UTC, +0000)</current-timezone>
</time>
```

Platform Details API

Table 16: Platform Details APIs

Action	Method	Payload Required	API
To get information about the hardware	GET	No	/api/operational/platform-detail

Sample Output for the Platform Details API

```
curl -k -v -u admin:Cisco123# -X GET 'https://172.19.162.209/api/operational/platform-detail'
Note: Unnecessary use of -X or --request, GET is already inferred.
* Trying 172.19.162.209...
* Connected to 172.19.162.209 (172.19.162.209) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
* CAfile: /etc/pki/tls/certs/ca-bundle.crt
CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
* subject: CN=nfv
* start date: Aug 17 11:21:43 2017 GMT
* expire date: Aug 15 11:21:43 2027 GMT
* issuer: CN=nfv
* SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/operational/platform-detail HTTP/1.1
> Host: 172.19.162.209
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.50.1
> Accept: */*
>< HTTP/1.1 200 OK
< Server: nginx
< Date: Fri, 18 Aug 2017 13:21:47 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<


<platform-detail
 xmlns="http://www.cisco.com/nfvos/platform-info"
 xmlns:y="http://tail-f.com/ns/rest"
 xmlns:platform_info="http://www.cisco.com/nfvos/platform-info">
 <hardware_info>
  <Manufacturer>Cisco Systems Inc</Manufacturer>
  <PID>UCSC-C220-M4S</PID>
  <SN>FCH1924V2AH</SN>
  <hardware-version>74-12419-01</hardware-version>
  <UUID>663F3347-5499-0D49-A76E-533A4AA9C755</UUID>
  <Version>3.6.0-916</Version>
  <Compile_Time>Monday, August 07, 2017 [01:30:11 PDT]</Compile_Time>
  <CPU_Information>Intel(R) Xeon(R) CPU E5-2630 v3 @ 2.40GHz 8 cores</CPU_Information>
  <Memory_Information>65701956 kB</Memory_Information>
  <Disk_Size>1000.2 GB</Disk_Size>
  <CIMC_IP>NA</CIMC_IP>
 </hardware_info>
 <software_packages>
  <Kernel_Version>3.10.0-514.10.2.el7.x86_64</Kernel_Version>
  <QEMU_Version>1.5.3</QEMU_Version>
  <LibVirt_Version>2.0.0</LibVirt_Version>
  <OVS_Version>2.3.2</OVS_Version>
 </software_packages>
 <port_detail>
  <Name>eth0</Name>
 </port_detail>
 <port_detail>
  <Name>eth1</Name>
 </port_detail>
 <port_detail>
  <Name>eth2</Name>
 </port_detail>
 <port_detail>
  <Name>eth3</Name>
 </port_detail>
 <port_detail>
  <Name>eth4</Name>
 </port_detail>
 <port_detail>
  <Name>eth5</Name>
 </port_detail>
 <switch_detail>
  <UUID>NA</UUID>
  <Type>NA</Type>
  <Name>NA</Name>
  <Ports>8</Ports>
 </switch_detail>
</platform-detail>
```

Port Details APIs

Table 17: Port Details APIs

Action	Method	Payload Required	API
To get information about the physical port	GET	No	/api/operational/platform-detail/port_detail

Sample Output for the Port Details API

```
curl -k -v -u admin:Cisco123# -X GET
'https://172.19.162.209/api/operational/platform-detail/port_detail'
Note: Unnecessary use of -X or --request, GET is already inferred.
* Trying 172.19.162.209...
* Connected to 172.19.162.209 (172.19.162.209) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
* CAfile: /etc/pki/tls/certs/ca-bundle.crt
CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
* subject: CN=nfv
* start date: Aug 17 11:21:43 2017 GMT
* expire date: Aug 15 11:21:43 2027 GMT
* issuer: CN=nfv
* SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/operational/platform-detail/port_detail HTTP/1.1
> Host: 172.19.162.209
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.50.1
> Accept: */*
>< HTTP/1.1 200 OK
< Server: nginx
< Date: Fri, 18 Aug 2017 13:24:32 GMT
< Content-Type: application/vnd.yang.collection+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<

<collection
 xmlns:y="http://tail-f.com/ns/rest">
 <port_detail
  xmlns="http://www.cisco.com/nfvos/platform-info">
  <Name>eth0</Name>
  <Type>physical</Type>
  <Media>Twisted Pair</Media>
  <Link>up</Link>
  <Speed>1000</Speed>
  <MTU>1500</MTU>
  <MAC>80:e0:1d:4a:8c:56</MAC>
  <PCI_detail>01:00.0</PCI_detail>
 </port_detail>
 <port_detail
  xmlns="http://www.cisco.com/nfvos/platform-info">
  <Name>eth1</Name>
  <Type>physical</Type>
  <Media>Twisted Pair</Media>
  <Link>up</Link>
  <Speed>1000</Speed>
  <MTU>1500</MTU>
  <MAC>80:e0:1d:4a:8c:57</MAC>
  <PCI_detail>01:00.1</PCI_detail>
 </port_detail>
 <port_detail
  xmlns="http://www.cisco.com/nfvos/platform-info">
  <Name>eth2</Name>
  <Type>physical</Type>
  <Media>Twisted Pair</Media>
  <Link>down</Link>
  <Speed>0</Speed>
  <MTU>1500</MTU>
  <MAC>80:e0:1d:37:0f:28</MAC>
  <PCI_detail>04:00.0</PCI_detail>
 </port_detail>
 <port_detail
  xmlns="http://www.cisco.com/nfvos/platform-info">
  <Name>eth3</Name>
  <Type>physical</Type>
  <Media>Twisted Pair</Media>
  <Link>down</Link>
  <Speed>0</Speed>
  <MTU>1500</MTU>
  <MAC>80:e0:1d:37:0f:29</MAC>
  <PCI_detail>04:00.1</PCI_detail>
 </port_detail>
 <port_detail
  xmlns="http://www.cisco.com/nfvos/platform-info">
  <Name>eth4</Name>
  <Type>physical</Type>
  <Media>Twisted Pair</Media>
  <Link>down</Link>
  <Speed>0</Speed>
  <MTU>1500</MTU>
  <MAC>80:e0:1d:37:0f:2a</MAC>
  <PCI_detail>04:00.2</PCI_detail>
 </port_detail>
 <port_detail
  xmlns="http://www.cisco.com/nfvos/platform-info">
  <Name>eth5</Name>
  <Type>physical</Type>
  <Media>Twisted Pair</Media>
  <Link>down</Link>
  <Speed>0</Speed>
  <MTU>1500</MTU>
  <MAC>80:e0:1d:37:0f:2b</MAC>
  <PCI_detail>04:00.3</PCI_detail>
 </port_detail>
</collection>
```

Portal Access APIs

Table 18: Portal Access APIs

Action	Method	Payload Required	API
To enable or disable the portal access	PUT	Yes	/api/config/system/portal
To get the portal access status	GET	No	/api/operational/system/portal/status

Example for a Portal Access Payload

```
<portal>
    <access>enabled</access>
</portal>
```

Table 19: Portal Access Payload Description

Property	Type	Description	Mandatory/Default Value
access	String	Specify the portal access as "enabled" or "disabled".	Yes

Example: PUT Portal Access (Enable/Disable)

```
curl -v -k -u "admin:Cisco123#" -H "Content-Type:application/vnd.yang.data+xml" -X
PUT https://209.165.201.1/api/config/system/portal -d
"<portal><access>enabled</access></portal>"

*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Mar 14 06:53:22 2017 GMT
*  expire date: Mar 12 06:53:22 2027 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> PUT /api/config/system/portal HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.50.1
> Accept: */*
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 41
>
* upload completely sent off: 41 out of 41 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.10.1
< Date: Tue, 14 Mar 2017 19:34:42 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Last-Modified: Tue, 14 Mar 2017 19:34:42 GMT
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Etag: 1489-520082-470197
< Pragma: no-cache
```

Example: GET Portal Access API

```
curl -v -k -u  admin:Cisco123# -X GET
'https://209.165.201.1/api/operational/system/portal/status'

Note: Unnecessary use of -X or --request, GET is already inferred.
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.2 (OUT), TLS handshake, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Server hello (2):
* NPN, negotiated HTTP1.1
* TLSv1.2 (IN), TLS handshake, Certificate (11):
* TLSv1.2 (IN), TLS handshake, Server key exchange (12):
* TLSv1.2 (IN), TLS handshake, Server finished (14):
* TLSv1.2 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.2 (OUT), TLS change cipher, Client hello (1):
* TLSv1.2 (OUT), TLS handshake, Unknown (67):
* TLSv1.2 (OUT), TLS handshake, Finished (20):
* TLSv1.2 (IN), TLS change cipher, Client hello (1):
* TLSv1.2 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.2 / ECDHE-RSA-AES256-GCM-SHA384
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Mar 14 06:53:22 2017 GMT
*  expire date: Mar 12 06:53:22 2027 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/operational/system/portal/status HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.50.1
> Accept: */*
>
< HTTP/1.1 200 OK
< Server: nginx/1.10.1
< Date: Tue, 14 Mar 2017 19:35:05 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
```

System Log APIs

Action	Method	Payload Required	API
To set system logs	POST	Yes	/api/operations/system/set-log
To get the system log configuration details	GET	No	/api/operational/system/logging-level

Example for System Log Payload

```
<input>
    <logtype>all</logtype>
    <level>warning</level>
</input>
```

Table 20: Payload Description for Setting Log Level

Property	Type	Description	Mandatory/Default Value
logtype	String	Type of the log. There are two types: configuration and operational. You can specify one of the following: • configuration • operational • all (includes both configuartion and opeartional logs)	Yes
level	String	Indicates the log level. The supported log levels are: debug, info, warning, error, and critcal. **Note** The info and warning log levels are set by default respectively for the configuration and operational log types. You can change them as required. However, the change to the log level is not persisted across a reboot. After a reboot, the default log levels are used.	Yes

Example: POST System Log API

```
curl -k -v -u admin:Cisco123# -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
POST https://209.165.201.1/api/operations/system/set-log -d
'<input><logtype>all</logtype><level>warning</level></input>'
*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.0 (OUT), TLS handshake, Client hello (1):
* TLSv1.0 (IN), TLS handshake, Server hello (2):
* TLSv1.0 (IN), TLS handshake, Certificate (11):
* TLSv1.0 (IN), TLS handshake, Server key exchange (12):
* TLSv1.0 (IN), TLS handshake, Server finished (14):
* TLSv1.0 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.0 (OUT), TLS change cipher, Client hello (1):
* TLSv1.0 (OUT), TLS handshake, Finished (20):
* TLSv1.0 (IN), TLS change cipher, Client hello (1):
* TLSv1.0 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.0 / DHE-RSA-AES256-SHA
* Server certificate:
```

```
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Dec  8 07:50:20 2016 GMT
*  expire date: Dec  6 07:50:20 2026 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> POST /api/operations/system/set-log HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.49.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
> Content-Length: 59
>
* upload completely sent off: 59 out of 59 bytes
< HTTP/1.1 204 No Content
< Server: nginx/1.6.3
< Date: Thu, 05 Jan 2017 03:49:32 GMT
< Content-Type: text/html
< Content-Length: 0
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
```

Example: GET System Log API

```
curl -k -v -u admin:Cisco123# -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X
GET https://209.165.201.1/api/operational/system/logging-level

*   Trying 209.165.201.1...
* Connected to 209.165.201.1 (209.165.201.1) port 443 (#0)
* Cipher selection: ALL:!EXPORT:!EXPORT40:!EXPORT56:!aNULL:!LOW:!RC4:@STRENGTH
* successfully set certificate verify locations:
*   CAfile: /etc/pki/tls/certs/ca-bundle.crt
  CApath: none
* TLSv1.0 (OUT), TLS handshake, Client hello (1):
* TLSv1.0 (IN), TLS handshake, Server hello (2):
* TLSv1.0 (IN), TLS handshake, Certificate (11):
* TLSv1.0 (IN), TLS handshake, Server key exchange (12):
* TLSv1.0 (IN), TLS handshake, Server finished (14):
* TLSv1.0 (OUT), TLS handshake, Client key exchange (16):
* TLSv1.0 (OUT), TLS change cipher, Client hello (1):
* TLSv1.0 (OUT), TLS handshake, Finished (20):
* TLSv1.0 (IN), TLS change cipher, Client hello (1):
* TLSv1.0 (IN), TLS handshake, Finished (20):
* SSL connection using TLSv1.0 / DHE-RSA-AES256-SHA
* Server certificate:
*  subject: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  start date: Dec  8 07:50:20 2016 GMT
*  expire date: Dec  6 07:50:20 2026 GMT
*  issuer: CN=Cisco-Enterprise-NFVIS-Self-Signed-Certificate
*  SSL certificate verify result: self signed certificate (18), continuing anyway.
* Server auth using Basic with user 'admin'
> GET /api/operational/system/logging-level HTTP/1.1
> Host: 209.165.201.1
> Authorization: Basic YWRtaW46Q2lzY28xMjMj
> User-Agent: curl/7.49.1
> Accept:application/vnd.yang.data+xml
> Content-Type:application/vnd.yang.data+xml
>
```

```
< HTTP/1.1 200 OK
< Server: nginx/1.6.3
< Date: Thu, 05 Jan 2017 03:45:53 GMT
< Content-Type: application/vnd.yang.data+xml
< Transfer-Encoding: chunked
< Connection: keep-alive
< Cache-Control: private, no-cache, must-revalidate, proxy-revalidate
< Pragma: no-cache
<
<logging-level xmlns="http://www.cisco.com/nfv" xmlns:y="http://tail-f.com/ns/rest"
xmlns:system="http://www.cisco.com/nfv">
  <configuration>info</configuration>
  <operational>warning</operational>
</logging-level>
```

DPDK Support APIs

Action	Method	Payload Required	API
To enable DPDK and VM migration	POST	Yes	/api/config/system/settings/
To Disable DPDK (in error state)	DELETE	No	/api/config/system/settings/dpdk
To get the status of DPDK	GET	No	/api/operational/system/settings-native/dpdk-status

Table 21: Payload Description for DPDK Support

Property	Type	Description	Mandatory/Default Value
dpdk	String	Specify enabling DPDK	Yes

Example : POST to enable DPDK

```
curl -k -v -u admin:admin -H "Accept:application/vnd.yang.data+json" -H
"Content-Type:application/vnd.yang.data+json" -X POST
https://localhost/api/config/system/settings/
--data '{"dpdk": "enable"}'
```

Example: DELETE to disable DPDK

```
curl -k -v -u admin:admin -X DELETE https://localhost/api/config/system/settings/dpdk
```

Example: GET to get the status of DPDK:

```
curl -k -v -u admin:admin -X GET
https://localhost/api/operational/system/settings-native/dpdk-status
```

Backup and Restore APIs

Backup APIs

Action	Method	Payload Required	API
To start configuration-only backup	POST	Yes	/api/operations/hostaction/backup/configuration-only/
To start configuration-and-vms backup	POST	Yes	/api/operations/hostaction/backup/configuration-and-vms/

Table 22: Payload Description for Setting Log Level

Property	Type	Description	Mandatory/Default Value
file-path	String	Path representing location to the file	Yes

Example: POST to start a configuration-only backup

```
curl -k -v -u admin:admin -H "Accept:application/vnd.yang.data+json" -H
"Content-Type:application/vnd.yang.data+json" -X POST
https://localhost/api/operations/hostaction/backup/configuration-only/
--data '{"input": {"file-path": "intdatastore:sample.bkup"}}'
```

Example: POST to start configuration-and-vms backup:

```
curl -k -v -u admin:admin -H "Accept:application/vnd.yang.data+json" -H
"Content-Type:application/vnd.yang.data+json" -X POST
https://localhost/api/operations/hostaction/backup/configuration-and-vms/
--data '{"input": {"file-path": "intdatastore:sample.bkup"}}'
```

Restore APIs

Action	Method	Payload Required	API
To start restore from a backup package	POST	Yes	/api/operations/hostaction/restore/

Table 23: Payload Description for Setting Log Level

Property	Type	Description	Mandatory/Default Value
restore-option	String	Option to restore without connectivity settings. Accepted values: except-connectivity	No

Example: To start a restore

```
curl -k -v -u admin:admin -H "Accept:application/vnd.yang.data+json" -H
"Content-Type:application/vnd.yang.data+json" -X POST
https://localhost/api/operations/hostaction/restore/
--data '{"input": {"file-path": "intdatastore:sample.bkup"}}'
```

Example: To start a restore while preserving connectivity settings:

```
curl -k -v -u admin:admin -H "Accept:application/vnd.yang.data+json" -H
"Content-Type:application/vnd.yang.data+json" -X POST
https://localhost/api/operations/hostaction/restore/
--data '{"input": {"restore-option": "except-connectivity", "file-path":
"intdatastore:sample.bkup"}}'
```

Route Distribution APIs

Action	Method	Payload Required	API
To configure route distribution	POST	Yes	/api/config/route-distributions
To update route distribution configuration	GET	No	/api/config/route-distributions?deep
To delete route distribution configuration	DELETE	No	/api/config/route-distributions
To get route distribution state data	GET	No	/api/operational/route-distributions

Example for route distribution payload

```
<route-distribute>
 <neighbor-address>172.25.221.106</neighbor-address>
 <local-bridge>wan-br</local-bridge>
 <local-as>65000</local-as>
 <remote-as>65000</remote-as>
 <network-subnet>
  <subnet>10.20.0.0/24</subnet>
 </network-subnet>
</route-distribute>
```

Table 24: Payload Description for Route Distribution

Property	Type	Description	Mandatory/Default Value
neighbor-address	String	Neighbor IPv4 address secure overlay connection.	Yes
local-address	String	Local IPv4 address	No
local-bridge	String	Local bridge name for overlay (default wan-br)	No

local-as	String	Local autonomous system number	Yes
remote-as	String	Remote autonomous system number	Yes
router-id	String	Local router id IP address	No
network-subnet	String	List of subnets to be announced. H.H.H.H/N (atleast one subnet needs to be announced)	Yes
next-hop	String	IPv4 address of any local interface	No

Example: POST create route distribution

```
curl -k -v -u "admin:admin" -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X POST
https://209.165.201.1/api/config/route-distributions -d
'<route-distribute><neighbor-address>172.25.221.106</neighbor-address><local-bridge>wan-br</local-bridge><local-as>65000</local-as><remote-as>65000</remote-as><network-subnet><subnet>10.20.0.0/24</subnet></network-subnet></route-distribute>'
```

Example: GET update route distribution

```
curl -k -v -u "admin:admin" -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X PUT
https://209.165.201.1/api/config/route-distributions/route-distribute/172.25.221.106 -d
'<route-distribute><neighbor-address>172.25.221.106</neighbor-address><local-bridge>wan-br</local-bridge><local-as>65000</local-as><remote-as>65000</remote-as><network-subnet><subnet>10.20.0.0/24</subnet></network-subnet></route-distribute>'
```

Example: GET route distributions state information

```
curl -k -v -u "admin:admin" -X GET
"https://209.165.201.1/api/operational/route-distributions?deep"
```

Example: DELETE all route distributions

```
curl -k -v -u "admin:admin" -X DELETE "https://209.165.201.1/api/config/route-distributions"
```

Dynamic SR-IOV APIs

Action	Method	Payload Required	API
To enable SR-IOV	PUT	Yes	/api/config/pnics/pnic/eth0-1/sriov/numvfs --
To set switchmode	PUT	No	/api/config/pnics/pnic/eth0-1/sriov/switchmode
To disable SR-IOV	DELETE	No	/api/config/pnics/pnic/eth0-1/sriov
To get SR-IOV operational data	GET	No	/api/operational/pnics/pnic/eth0-1/sriov
To create SR-IOV network with trunk mode	POST	Yes	/api/config/networks

Action	Method	Payload Required	API
To create SR-IOV network with access mode	POST	Yes	/api/config/networks
To delete SR-IOV network	DELETE	No	/api/config/networks/network/eth0-1-SRIOV-1

Example: PUT enable SR-IOV

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X PUT
https://209.165.201.1/api/config/pnics/pnic/eth0-1/sriov/numvfs --data '<numvfs>1</numvfs>'
```

Example: DELETE disable SR-IOV

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X DELETE
https://209.165.201.1/api/config/pnics/pnic/eth0-1/sriov
```

Example: GET SR-IOV operational data

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X GET
https://209.165.201.1/api/operational/pnics/pnic/eth0-1/sriov
```

Example: POST create SR-IOV network with trunk mode

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X POST https://209.165.201.1/api/config/networks
 --data '<network><name>eth0-1-SRIOV-1</name><sriov>true</sriov></network>'
```

Example: POST create SR-IOV network with access mode

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X POST https://209.165.201.1/api/config/networks
 --data
'<network><name>eth0-1-SRIOV-1</name><sriov>true</sriov><trunk>false</trunk><vlan>30</vlan></network>'
```

Example: DELETE SR-IOV network

```
curl -k -v -u admin:admin -H Accept:application/vnd.yang.data+xml -H
Content-Type:application/vnd.yang.data+xml -X DELETE
https://209.165.201.1/api/config/networks/network/eth0-1-SRIOV-1
```